ELETTRONICA

LA RIVISTA DI ELETTRONICA APPLICATA, TECNOLOGIE E PRODOTTI

APRILE 2005 - ANNO 21 - N° 238

€ 5,50

Poste Italiane Spa - Spedizione in abbonamento Postale - D.L. 353/2003 (conv. in L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano.
In caso di mancato recapito, restituire all'editore che si impegna a pagare la relativa tassa presso il CPM di Roserio - Milano

NOVITÁ

CORSO DI PROGRAMMAZIONE IN BASIC PER PICMICRO®

CD-ROM ALLEGATO

- Un database di oltre 150.000 dispositivi
- Tutti i codici di identificazione per package SMD
- Interfaccia e guida in linea interamente in italiano
- Tutte le dimensioni meccaniche dei packages
- Un CD appositamente creato per i lettori di Fare Elettronica

ELETTRONICANDO

4ª parte: I semiconduttori e il diodo pn

PRATICA

I Fotoaccoppiatori:
Avvisatore ottico di chiamata telefonica

Il sistema DCC

Il booster, il modulo di servizio e il software di controllo

TEORIA

Intelligenza Artificiale

Introduzione alle reti neurali

Alimentatori switching

Il controllo PWM negli alimentatori switching

SPECIALE

I radar ad apertura sintetica

www.farelettronica.com

INWARE EDIZIONI

www.farelettronica.com

DIRETTORE RESPONSABILE:
Antonio Cirella

DIRETTORE DI REDAZIONE:
Tiziano Galizia

DIRETTORE TECNICO:
Maurizio Del Corso

HANNO COLLABORATO IN QUESTO NUMERO:
Maurizio Del Corso, Giovanni Di Maria, Riccardo Nicoletti, Antonio Di Stefano, Corrado D'Achille, Raffaele Nutricato, Romano Bernarducci, Nico Grilloni.

DIREZIONE - REDAZIONE - PUBBLICITÁ:
INWARE srl - Via Cadorna, 27/31
20032, Cormano (MI)
Tel. 02.66504794 - 02.66504755
Fax 02.66508225
info@inware.it - www.inwaredizioni.it

PROGETTO GRAFICO E IMPAGINAZIONE:
Graficonsult - Milano

STAMPA:
ROTO 2000 - Via L. da Vinci, 18/20
20080, Casarile (MI)

DISTRIBUZIONE:
Parrini & C. S.p.a. - Viale Forlanini, 23
20134, Milano

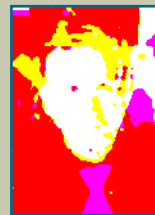
UFFICIO ABBONAMENTI
PARRINI & C. S.p.a.
Servizio abbonamenti
Viale Forlanini, 23 - 20134 Milano
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
abbonamenti@farelettronica.com
Tel. 02.66504794 - Fax. 02.66508225

Poste Italiane S.p.a.
Spedizione in abbonamento Postale D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano.
Abbonamento per l'Italia: € 45,00
Abbonamento per l'estero: € 115,00

Gli arretrati potranno essere richiesti, per iscritto, al seguente costo:
Numero singolo: € 7,50
Numero doppio: € 9,00
Numero con allegato: € 8,50

Autorizzazione alla pubblicazione del Tribunale di Milano n. 647 del 17/11/2003.
Iscrizione al R.O.C. n. 11035 19/11/2003

Note dalla redazione

Tiziano Galizia
t.galizia@farelettronica.com

Paolo, per gli amici "Kit"...

Paolo era un ragazzo occhialuto esperto di elettronica, uno di quelli che oggi in un simpatico gergo informatico chiameremmo "nerd".
Lavorava come commesso nel negozio di elettronica vicino casa mia. Lui ti salutava sempre con un sorriso e, dentro la tasca del suo camice bianco, portava un libricino con tutte le equivalenze per gli IC ed i transistor allora in produzione. Solo per passione riparava i kit che avevamo montato e, con nostra grande delusione, non avevano funzionato, ma quello per cui Paolo era famoso è il "kitting" (da qui il soprannome "Kit"). In pratica, prima di realizzare un progetto andavamo da lui ed, elenco componenti alla mano, cercavamo di convincerlo a darci tutti i componenti, promettendogli che sarebbe stata l'ultima volta che gli facevamo perdere così tanto tempo. Paolo con pazienza certosina, nella pausa pranzo, si fermava in negozio e ci preparava il kit, arrivando addirittura a sostituire i componenti mancanti con dei tipi compatibili o combinando resistenze e condensatori in modo da raggiungere il valore desiderato. Poi nel pomeriggio andavamo a ritirarli e li trovavamo divisi per tipologia in tante bustine di plastica trasparente.
Proprio il mese scorso per una serie di coincidenze sono tornato in quel negozietto dopo più di venti anni, non mi aspettavo di trovarlo uguale ci mancherebbe altro, ma trovarlo trasformato in un centro per la TV satellitare mi ha fatto un po' di tristezza.
Ho dato uno sguardo in giro alla ricerca di elementi familiari, niente, completamente diverso, poi ho chiesto se Paolo lavorasse ancora li e un commesso sulla trentina, neanche troppo gentile, mi ha risposto che non ne aveva mai sentito parlare.
Vi ho raccontato di *Paolo Kit* non tanto per analizzare i motivi per cui molti negozi di componenti elettronici hanno chiuso o si sono convertiti ad attività più remunerative, quanto per ricordare un modo di fare elettronica che si è inesorabilmente arreso alla velocità e flessibilità di Internet ma che fa parte del DNA di molti di noi.

State per leggere un numero davvero speciale, non solo per il bellissimo CD in allegato o per le 16 pagine in più, ma anche per i tanti articoli interessanti che lo compongono tra i quali, come avete potuto notare dalla copertina, il corso sul *Mikrobasic*, un compilatore Basic per PICmicro che è possibile scaricare gratuitamente in versione demo dal nostro sito e, anche se limitato nella lunghezza del codice, vi consentirà di fare esperienza.
A questo punto non mi resta che augurarvi una piacevole lettura e darvi appuntamento in edicola a Maggio.

Note dall'editore

Antonio Cirella
a.cirella@farelettronica.com

L'importanza di un evento

Come in ogni "periodo di fiere" che si rispetti, anche in questo ci si interroga se sia ancora il caso di partecipare ai vari eventi, se questi possano ancora darci "quel valore aggiunto" che ci aspettiamo oppure ci riserveranno solo delusione.
La stragrande maggioranza dei visitatori di un evento infatti intende ottenere in maniera diretta un contatto col mercato, cercare informazioni importantissime e vitali per il proprio lavoro o hobby che sia. È altresì molto importante anche la motivazione di cercare nuovi partner, nuovi prodotti o idee, stabilire nuove, proficue, relazioni. Marginale invece la necessità di "tenere sotto controllo" la concorrenza, motivazione principe dei tempi che furono. In pratica c'è sempre un motivo fondamentale che spinge il visitatore a recarsi personalmente alla manifestazione, è proprio questa motivazione che dovrebbe fornire il sufficiente incoraggiamento agli espositori per "tener duro" alle lusinghe di altre metodologie marketing sostitutive. Molte aziende infatti praticano la strategia del Roadshow, che gli permette di raggiungere clienti attuali e potenziali con giri itineranti auto-organizzati.
Questa strategia presenta diversi vantaggi: nessun concorrente presente, il partecipante non deve spostarsi dato che il Roadshow lo raggiunge nella sua città. Tuttavia, la manifestazione classica, organizzata da un ente organizzatore esterno, è e rimane il vero "contatto col mercato" in quanto dall'eterogeneità degli espositori si trae uno spaccato dell'offerta, mentre l'eterogeneità dei visitatori garantisce un ottimale spaccato della domanda.
Altro particolare fondamentale, ogni organizzatore permette la realizzazione di convegni solitamente molto frequentati ed interessanti, occasione per i progettisti di prendersi della formazione utilissima per il proprio lavoro.
Il Roadshow, il Workshop, e gli altri eventi similari, restano degli strumenti marketing molto efficaci, ma che poco hanno a che fare con le manifestazioni fieristiche, garanzia di pluralità e innovazione, preziosissime entrambi per un settore ad alta competizione come quello della componentistica elettronica.

Richieste di assistenza

Per richiedere assistenza o chiarimenti sugli articoli pubblicati, vi preghiamo di contattare l'autore, il cui nome ed indirizzo email è sempre riportato sotto il titolo dell'articolo stesso.
Nel caso ciò non fosse possibile potete scrivere a mailbox@farelettronica.com, ricordandovi di specificare il numero della rivista ed il titolo dell'articolo per il quale chiedete chiarimenti, oltre al vostro nome, cognome ed indirizzo e-mail.
Tutte le richieste con informazioni insufficienti o anonime non saranno prese in considerazione.

Collaborare con Fare Elettronica

La redazione di Fare Elettronica è alla ricerca di collaboratori per la stesura di articoli, progetti, tutorials, rubriche e libri.
Le richieste di collaborazione vanno indirizzate a Tiziano Galizia *(t.galizia@farelettronica.com)* e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

Come contattarci

INWARE Edizioni
Via Cadorna, 27/31
20032 Cormano (MI)
Telefono 02.66504794
Fax 02.66508225

Redazione:
redazione@farelettronica.com

Ufficio Abbonamenti:
abbonamenti@farelettronica.com

Elenco inserzionisti

Sommario

238
Aprile
2005

Pratica

pag. 34

Teoria

pag. 20

Risorse

pag. 52

Speciale

VRT-DISK 2005

Un'esclusiva per i lettori di Fare Elettronica: ECA vrt-disk 2005 limited edition. Un database di oltre 150.000 dispositivi a semiconduttore, per ciascuno dei quali sono riportate le principali caratteristiche elettriche e meccaniche ed altre utilissime informazioni. Ma scopriamolo più in dettaglio...

VRT-DISK ECA 2005

Chi opera nel settore elettronico si è sicuramente trovato a sfogliare gli storici "libri gialli" i cui protagonisti non assomigliano certo ai mitici *Sherlok Holmes* o *Hercule Poirot*, ma sono nientemeno che componenti elettronici a semiconduttore! Stiamo parlando dei libri ECA, rigorosamente presenti in ogni negozio di componenti elettronici dapprima nel caratteristico formato "stretto e lungo", poi nel formato B5 ed infine nel formato elettronico, su CD-ROM, meno ingombrante e molto più facile da usare. Vista la loro grande utilità Fare Elettronica, in collaborazione con ECA, ha realizzato per voi una versione speciale del **ECA vrt-disk 2005**, interamente in italiano che risulterà assai utile a tutti voi lettori.

IL CONTENUTO DEL CD-ROM

Il CD-ROM contiene un database di oltre 150.000 dispositivi a semiconduttore tra cui diodi, transistor, SCR e circuiti integrati oltre alla tabella dei codici SMD e gli ingombri meccanici dei package. Il database è una versione limitata per cui, per ciascun dispositivo, sono mostrate solo le caratteristiche essenziali, ma senz'altro utili. Un potente motore di ricerca consente di selezionare i dispositivi desiderati grazie anche alla possibilità di utilizzare i caratteri jolly (come * o ?) nei criteri di ricerca.

L'INSTALLAZIONE

Il processo di installazione è totalmente automatico: dopo aver accettato il contratto di licenza, si sceglie il percorso di installazione quindi si avvia l'installazione confermando tutte le eventuali finestre mostrate. Ad installazione completata, sarà possibile avviare il programma cliccando sulla relativa icona creata sul desktop o dal nuovo collegamento creato nel gruppo *Programmi*. All'avvio, l'interfaccia utente si presenta come in figura 1.

Figura 1 *L'interfaccia utente del database ECA*

LE INFORMAZIONI DISPONIBILI

Come si vede dalla figura 1, ogni componente è caratterizzato da diversi campi, ciascuno dei quali fornisce diverse informazioni relative al componente selezionato. Ecco la descrizione dettagliata del contenuto di ciascun campo.

- *Tipo:* È la sigla con cui viene identificato il com-

il CD-ROM allegato

ponente (ad esempio BC109, 7410, 4001).

- *Dispositivo:* Riporta la tecnologia con cui è costruito il componente ed eventualmente la funzione svolta (ad esempio GaAs indica un dispositivo all'arseniuro di gallio, *MOS-N-FET-d* indica un MOSFET a canale N a svuotamento).
- *Descrizione:* Questo campo contiene utili informazioni relative al componente. Ad esempio Gi indica che il dispositivo è un raddrizzatore, il simbolo "=" seguito da un codice significa che il componente selezionato e quello indicato sono equivalenti, *E* indica uno stadio di uscita e così via. Nella guida in linea sono riportate tutte le abbreviazioni usate per questo campo.
- *Case:* Contiene informazioni sul package del componente. 18-DIP indica ad esempio un package a 18 pin dual-in-line oppure TO18 per il classico contenitore a tre pin per i transistor.
- *Produttore:* È il campo che identifica il produttore del componente (ad esempio Fch per Fairchild, Hit per Hitachi). L'elenco di tutti i codici identificativi dei costruttori sono elencati nella guida in linea.
- *ECA-book:* È un riferimento ai database dei prodotti ECA in cui è incluso il componente.

Con un doppio click sul nome di un dispositivo, si accede alla scheda tecnica (in figura 2 la scheda per il transistor 2N1711) nella quale sono riassunte le informazioni sul componente. Alcune informazioni sono disponibili solo nella versione completa.

IL MOTORE DI RICERCA

Uno strumento indispensabile per l'individuazione di uno specifico componente o di un gruppo di componenti che hanno determinate caratteristiche comuni, è il motore di ricerca. Utilizzando i pulsanti indicati nella figura 3, è possibile effettuare una ricerca per tipo, per dispositivo o per contenitore.
Nel primo caso dovrà essere specificata la sigla di un componente (ad es. 7400) ed è possibile utilizzare i caratteri jolly "?" e "*" per cui effettuando una ricerca per tipo indicando 74*, verranno visualizzati tutti i componenti il cui campo *tipo* inizia per 74.
In modo analogo si può fare una ricerca per dispositivo (ad esempio ricercando *Gi* verranno elencati tutti i dispositivi raddrizzatori) o per tipo di contenitore (in questo caso specificando ad esempio *TO** verranno elencati tutti i componenti in contenitore TO).

BUON LAVORO!

Il CD-ROM ECA sarà senz'altro utile a tutti gli appassionati e professionisti dell'elettronica. Seguendo le istruzioni riportate nel CD, è possibile acquistare la versione completa usufruendo di uno sconto speciale riservato ai lettori di Fare Elettronica. L'offerta è valida fino al 1.7.2005. Un'occasione da prendere al volo!

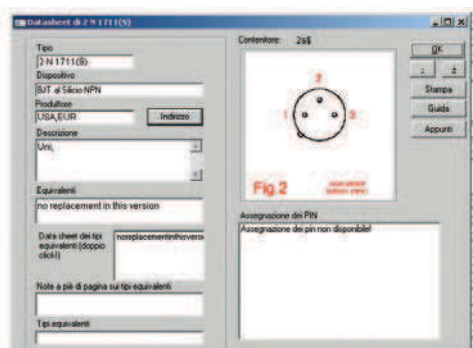



Figura 2 *Le informazioni per il 2N1711*

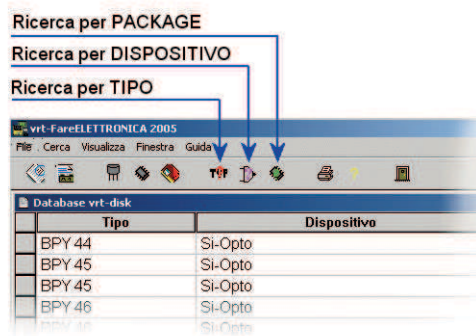



Figura 3 *I diversi tipi di ricerca*

Scrivete a:
MAILBOX
REDAZIONE DI
FARE ELETTRONICA
Inware s.r.l.
Via Cadorna, 27/31
20032, Cormano (MI)

Oppure inviate un'email a:
mailbox@farelettronica.com

Mailbox

Questa rubrica ospita le richieste più interessanti pervenute dai lettori. Per quanto possibile verrà data risposta a tutte le richieste pervenute via email.

MUSICASSETTA PER MP3

Mi è capitato tra le mani il numero 228 della vostra rivista (nella biblioteca della mia scuola) e vorrei chiedervi, riguardo all'articolo "ascoltiamo i file mp3 con la nostra vecchia autoradio stereo", qualche informazione in più.
Io ho provato a realizzare quella falsa musicassetta, ma quando l'ho inserita nella mia autoradio (anzi "aperadio") non è successo niente, solo un ronzio di sottofondo, anche se cambiavo lato!
Dove posso aver sbagliato? I due fili dell'uscita del lettore cd, li potevo mettere a caso nella testina magnetica?
Forse ho sbagliato la distanza a cui ho attaccato quest'ultima nella musicassetta?!

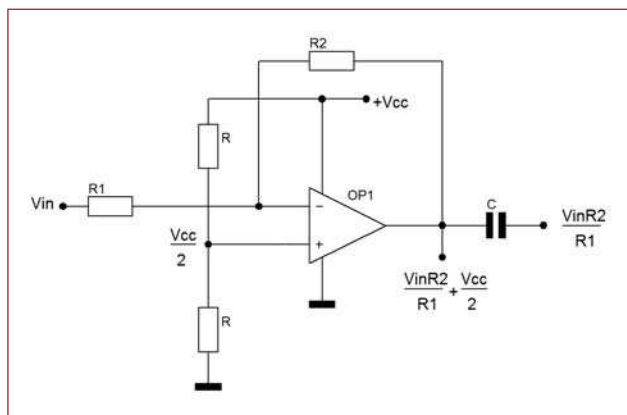
Andrea Urbinati

La parte forse più difficile è la taratura dell'altezza della testina che deve essere alloggiata nella finta cassetta. Per fare questo l'autore suggerisce di operare in questo modo:
1) Procurarsi uno di quei walkman che si usavano una volta e che hanno la caratteristica di potere asportare completamente il coperchio di protezione ed accedere quindi alla vite di regolazione dell'azimut (altezza verticale) della testina magnetica. Non importa che sia di alta qualità, deve solo essere funzionante.
2) Inserendo una musicassetta originale, premere play e quindi, ascoltando in cuffia quello che esce dal walkman, si deve regolare l'altezza della testina fino ad avere il suono stereo più limpido e bilanciato possibile. A questo punto abbiamo il nostro "strumento" di riferimento per potere tarare la nostra "finta cassetta". Infatti i riferimenti meccanici che posizionano le cassette i tutti i mangiacassette hanno le stesse dimensioni; tarando quindi opportunamente sul walkman il nostro trasduttore (la finta cassetta che abbiamo costruito) siamo certi che questa possa essere riprodotta anche dalla nostra autoradio.
3) Dobbiamo essere sicuri che i collegamenti siano fatti correttamente. La testina stereo normalmente ha tre reofori (connessioni): due che devono essere collegati al polo caldo dell'uscita cuffia di un qualsiasi apparato stereo e uno che rappresenta il collegamento a massa.
4) Posizionare il semiguscio inferiore nel walkman, inserire il jack nel lettore di mp3 e azionare il tasto play. In questo modo la potenza che dovrebbe essere trasmessa alla cuffia viene trasmessa alla testina magnetica. Al walkman colleghiamo invece una cuffia, regolando la potenza di uscita su un volume medio. Dovremo muovere la testina della nostra finta cassetta finché non otterremo un suono soddisfacente in cuffia. A questo punto con la colla termica la fissiamo subito, attendendo che si raffreddi.
5) Fissiamo meccanicamente i fili, chiudiamo la cassetta e tutto dovrebbe funzionare bene.
Come spiegato nell'articolo, alcuni mangianastri hanno delle logiche molto sofisticate, difficili da ingannare. Il trasduttore funziona egregiamente con le autoradio più spartane, che non hanno ricerca di brani, salto degli spazi vuoti, autoreverse automatico in base al silenzio e così via.

“Richieste, chiarimenti, dubbi e commenti dai lettori”

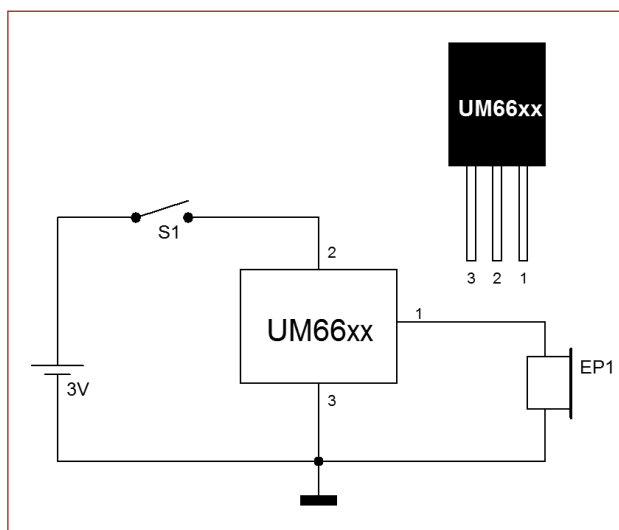
AMPLIFICATORE A SINGOLA ALIMENTAZIONE

Vorrei realizzare un amplificatore audio con un operazionale evitando l’utilizzo di una tensione di alimentazione duale.
Ho sentito parlare di masse flottanti o roba simile, ma sinceramente non ho capito molto.

Gianfranco Carloni

In un amplificatore operazionale, l’alimentazione duale permette di ottenere in uscita una gamma di tensioni comprese circa tra l’alimentazione positiva e quella negativa. Alimentando un operazionale solo con una tensione positiva verranno tagliate tutte le semionde negative del segnale in uscita provocando così forti distorsioni inammissibili specialmente in campo audio. L’astuzia che si usa in questi casi è quella di sommare al segnale di ingresso una tensione continua (normalmente pari alla metà della tensione di alimentazione) per fare in modo che il segnale di uscita non scenda mai al disotto dello zero. Lo zero dei segnali non sarà più il negativo di alimentazione, bensì il valore di tensione sommato all’ingresso (massa flottante). Per rimuovere la componente continua presente nel segnale di uscita sarà sufficiente inserire un condensatore in serie al carico in modo da lasciar passare solamente la componente dinamica della tensione. Utilizzando un amplificatore operazionale in configurazione invertente, sommare una componente continua al segnale di ingresso risulta particolarmente semplice in quanto è sufficiente collegare il morsetto non invertente alla tensione da sommare all’ingresso. In figura il partitore resistivo consente di ottenere una tensione di Vcc/2 che viene sommata al segnale di ingresso con la tecnica citata.

IC PER GENERARE MELODIE

Ho saputo che esistono dei circuiti integrati in grado di suonare specifiche melodie. Vorrei conoscere la sigla, come posso scegliere la melodia e in che modo utilizzarli.

Mauro Vigoretti

Gli integrati a cui si riferisce il lettore sono quelli della serie UM66T. Questi dispositivi, usati tipicamente per la realizzazione di campanelli per le abitazioni, hanno al loro interno una ROM in cui è memorizzata una specifica melodia. Dalla sigla del dispositivo si può risalire al tipo di melodia

memorizzata secondo la seguente tabella:

UM66T01:	Jingle Bells + Santa Claus is coming to town + We wish you a Merry X'mas
UM66T02:	Jingle Bells
UM66T04:	Jingle Bells + Rudolph, the red nosed Reindeer + Joy to the world
UM66T05:	Home sweet home
UM66T06:	Let me call you sweetheart
UM66T08:	Happy birthday to you
UM66T09:	Wedding March
UM66T11:	Love me tender, love me true
UM66T13:	Easter Parade
UM66T19:	For Elise
UM66T32:	Coo Coo Waltz
UM66T33:	Mary had a little limb
UM66T34:	The train is running fast
UM66T68:	It's a Small World

Nella figura, la più semplice applicazione impiegante un trasduttore piezoelettrico.

LUCE DI CORTESIA

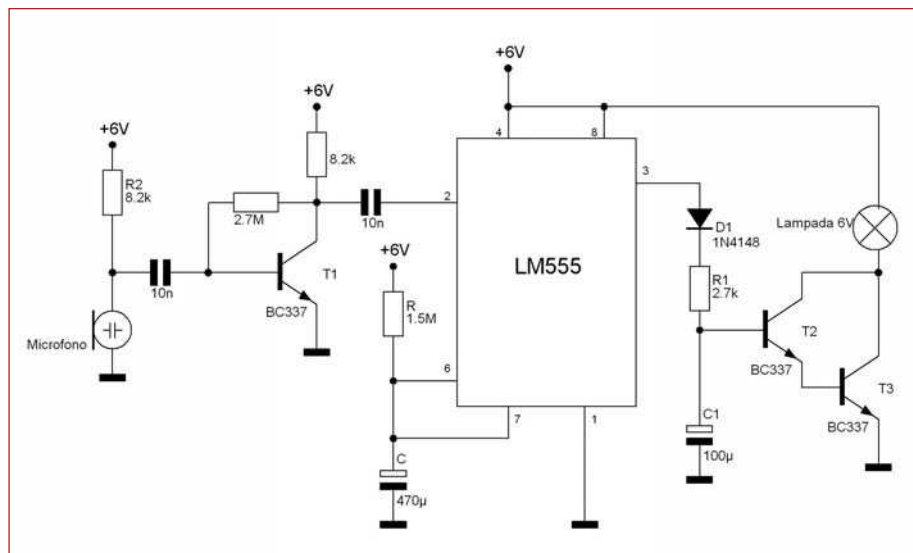
Spettabile Redazione, avrei necessità di realizzare un circuito alimentato a 6V che sia in grado di mantenere accesa una piccola lampadina in presenza di rumore e che la stessa venga spenta automaticamente dopo una decina di minuti di completo silenzio. Questo circuito permetterebbe a mio figlio di addormentarsi con la luce accesa senza preoccuparmi di spegnerla una volta che si è addormentato. Grazie per la vostra eventuale risposta.

Simone Faralli

Il circuito richiesto dal lettore può essere facilmente realizzato impiegando un timer 555 e pochi altri componenti esterni. Il circuito è quello riportato nella figura. Il segnale proveniente dal microfono piezoelettrico, viene amplificato dal transistor T1 collegato come amplificatore ad emettitore comune. Il segnale così ottenuto viene utilizzato per l'innesco del multivibratore monostabile realizzato con il timer 555. Il tempo di ritardo è determinato dalla coppia R e C e, con i valori indicati, il ritardo è di circa 12 minuti. La lampadina a 6V viene pilotata mediante la coppia di transistori T2 e T3 connessi in una configurazione Darlington. Fintanto che il microfono capta dei suoni esterni, il monostabile viene continuamente sollecitato, quindi la lampada risulta sempre accesa. In assenza di rumore la lampada viene spenta in maniera graduale grazie alla scarica piuttosto lenta di C1 che interdice gradualmente la coppia T2, T3. Il diodo impedisce che C1 si scarichi attraverso la resistenza R1 e l'uscita del timer.

Con la lampada spenta il consumo del circuito si aggira sui 5mA. Come lampada si può usare una comune lampadina come quelle usate nei fari delle biciclette.

La resistenza R2 regola la sensibilità del microfono: diminuendo il valore di R2 il microfono viene polarizzato con una corrente maggiore quindi diviene più sensibile.

Si consiglia comunque di non scendere al di sotto del kiloOhm per non danneggiare il microfono.

Sistemi di Registrazione Video

Videoregistratore per impianti di sicurezza in grado di registrare per un periodo massimo di 960 ore consecutive utilizzando una normale cassetta video da 180 minuti. Grazie alla sua meccanica di precisione, il TIME LAPSE consente di ridurre i frame al secondo memorizzati sul nastro arrivando a registrare un frame ogni 6 s circa pur mantenendo lo standard di registrazione PAL. In caso di mancanza di tensione di rete riprende a registrare al ripristino della stessa segnalando l'ora di interruzione e quella di ripartenza.

TIME LAPSE VCR 960 ore

FR138 320,00 Euro

TIME LAPSE 960 ore alimentazione 12VDC

Ideale per applicazioni su mezzi mobili, automobili, camion, barche e ovunque non sia disponibile la tensione di rete. La massima durata di registrazione è di 960 ore mentre in Real-Time (audio + video) la durata massima è di 30 ore. Il consumo contenuto (appena 15W) garantisce una buona autonomia anche nel caso di alimentazione con batterie di limitata capacità. Sistema di registrazione: 2 + 4 testine; Registrazione Audio: in modalità 3,6,18 e 30 ore; Velocità del nastro: 33,35 m/sec (modalità 3 ore); Videocassette: VHS; Tempi Registrazione/Riproduzione: 3, 6, 18, 30, 48, 72, 168, 240, 480, 720 960 ore; Sistema: PAL.

FR244 320,00 Euro

Registratore video per impianti di sicurezza con Hard-disk da 80 Gb incorporato. Dispone di quattro canali di ingresso con possibilità di registrazione anche in modalità QUAD. Velocità di registrazione massima di 25 fpc (100 frame/rate in modalità quad). Risoluzione di 640x224 pixel in registrazione e di 720x576 in visualizzazione. La durata massima di registrazione è compresa tra 30 ore circa (alta risoluzione, 30 fps) e oltre 1500 ore (bassa risoluzione, 1 fps).

FR222 780,00 euro

DVR 4 canali con HDD 80 Gb

Eccezionale DVR a 4 canali con hard disk estraibile e interfaccia Ethernet per connessione LAN/Internet. Questo apparecchio dispone inoltre di uscita USB per lo scarico veloce delle immagini senza rimozione dell'HDD. Le immagini riprese dalle quattro telecamere possono essere registrate su HDD e controllate da remoto tramite Internet; in pratica l'apparecchio funziona come un vero e proprio Video Web Server a 4 canali. Il DVR viene fornito senza hard disk.

FR261 580,00 euro

NETWORK DVR 4 canali con USB

DIGITAL VIDEO RECORDER

Funziona in maniera autonoma senza l'ausilio di un PC. Il DVR converte il segnale video proveniente dalla telecamera in immagini digitali che vengono salvate su un hard disk removibile. Al termine del periodo di registrazione l'Hard Disk può essere sostituito, cancellato oppure riscritto. DVR completamente autonomo; elevata qualità dell'immagine grazie alla compressione video di tipo Wavelet; ingresso video, compatibile con la maggior parte dei multiplexer; supporta sia lo standard PAL che quello NTSC; Il DVR viene fornito senza Hard Disk.

FR190 520,00 euro

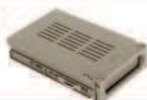
Funziona in maniera autonoma senza l'ausilio di un PC. Il DMR (Digital Multiplexer Recorder) converte il segnale video proveniente dalle telecamere d'ingresso in immagini digitali che vengono salvate su un hard disk removibile. Questo dispositivo integra in un unico apparecchio un registratore digitale video ed un multiplexer full duplex a 16 canali. Al termine del periodo di registrazione l'Hard Disk può essere sostituito, cancellato oppure riscritto. Formato video: NTSC/EIA O PAL/CCIR. Supporta 2 Hard Disk di tipo IDE (oltre 200 GB). Il DVR viene fornito senza Hard Disk.

FR233 960,00 euro

DVR con MULTIPLEXER 16 canali

CASSETTO PORTA HDD

Cassetto estraibile per hard disk adatto ai DVR FR190 e FR233.

DVRCARTR 32,00 euro

CASSETTO COMPLETO PER HDD

Sistema completo per hard disk estraibile composto da una slitta e da un cassetto estraibile. Adatto per i DVR FR190 e FR233 DVRCARTRIO 53,00 euro

FUTURA ELETTRONICA

Via Adige, 11 - 21013 Gallarate (VA) - Tel. 0331/799775 - Fax 0331/778112

Disponibili presso i migliori negozi di elettronica o nel nostro punto vendita di Gallarate (VA).
Caratteristiche tecniche e vendita on-line sul sito: www.futuranet.it

SCHEDE TECNICHE DETTAGLIATE: WWW.FUTURANET.IT

Tutti i prezzi sono da intendersi IVA inclusa.

Rubrica di notizie e novità dal mondo dell'elettronica.

Prima pagina

238-01 MICROCHIP INTEGRA I PROPRI MICROCONTROLLER PIC® CON BLOCCHI ANALOGICI

Microchip annuncia il nuovo PIC16F785, un nuovo dispositivo dotato di una serie di periferiche analogiche onboard. La dotazione prevede un A/D converter, dei comparatori, un riferimento di tensione e due amplificatori operazionali (op amps). L'equipaggiamento rende il dispositivo la soluzione ideale per le applicazioni più economiche di controllo digitale nei progetti di conversione dell'energia nonché nelle applicazioni di sensing e di controllo closed-loop. Il nuovo PIC16F785 da 20-pin permette di indirizzare in modo estremamente conveniente un ampio spettro di parametri di conversione dell'energia in applicazioni quali Switch Mode Power Supply (SMPS) e caricabatterie. I due amplificatori operazionali general-purpose offrono un livello di GBWP (Gain Bandwidth Product) di 3 MHz e una tensione di offset in ingresso di 5 millivolt. Tali valori sono particolarmente indicati per implementare filtri, stadi di guadagno e altre funzioni. I due op amp, unitamente al riferimento integrato da 1.2V e ai due comparatori ad alta velocità (40 nanosecondi), mettono a disposizione un livello d'integrazione senza rivali per quanto riguarda il sensing in ingresso. Tra le alter caratteristiche del PIC16F785 si segnalano: PWM a due fasi ad alta velocità con feedback asincrono - 3584 byte di memoria Flash programma, 256 byte di EEPROM e 128 byte di RAM - oscillatore interno di precisione a 8 MHz - tecnologia nanoWatt di power-management - 12 canali di A/D a 10-bit - Modulo Capture/Compare/PWM (CCP) - Reset brown out controllato via software e Watchdog Timer esteso.

238-02 DRIVER PER LED BIANCHI AD ELEVATA CORRENTE, IN MODALITÀ BUCK-BOOST

Linear Technology Corporation presenta il dispositivo LTC3453, un convertitore CC/CC buck-boost sincrono, ottimizzato per pilotare LED bianchi a corrente elevata, fino a 500 mA, da un ingresso di batteria a ioni di Litio. Il regolatore funziona automaticamente in modalità buck sincrona, boost sincrona o buck-boost, in funzione della tensione d'ingresso e della massima tensione diretta del LED. Ciò è di particolare importanza nelle applicazioni a LED a corrente elevata; poiché la tensione diretta del LED può variare da 2,7 V a 4,9 V, in funzione dei livelli di corrente e delle tolleranze del LED stesso.

La topologia buck-boost dell'LTC3453 garantisce un rendimento del 90% sull' intera gamma di alimentazione da 2,7 V a 4,2 V. Le regolazioni indipendenti di alta corrente e bassa corrente per le modalità flash e torcia/video possono essere rispettivamente programmate con facilità con 2 resistori per la regolazione di corrente e pin doppi di abilitazione.

Un rendimento del 90% è ottenuto per tutta la gamma tensione fornita da una batteria a ioni di Litio. La corrente del LED è programmabile da uno a quattro livelli (compreso l'arresto) con 2 resistori di regolazione di corrente e pin doppi di abilitazione. Nell'arresto, la corrente di alimentazione è di soli 6 µA.

Il dispositivo si autoprotegge in condizioni di LED aperto o chiuso. Un'alta frequenza, costante, di funzionamento, pari a 1 MHz, permette l'uso di un piccolo induttore esterno. L'LTC3453 è offerto in package QFN a 16 pin di mm 4 x 4 a basso profilo (0,75 mm), con ottime caratteristiche termiche.

PER SAPERNE DI PIÙ

Per approfondire le notizie riportate in questa rubrica, visitate il sito www.farelettronica.com/primapagina e seguite le istruzioni.
In alternativa potete scrivere a:

Inware Edizioni
Servizio Prima Pagina
Via Cadorna 27 - 20032 Cormano (MI)

Indicando il codice riportato accanto al titolo della notizia (esempio 238-03).

238-03 MICROCONTROLLORI USB LOW SPEED DELLA PROSSIMA GENERAZIONE

Cypress Semiconductor Corporation ha reso noto la disponibilità di una nuova famiglia di microcontrollori USB "low speed" da utilizzare in una vasta gamma di dispositivi di interfacciamento con i personal computer quali tastiere, mouse e "dongle" wireless. Basati sulla diffusa architettura del microcontrollore M8, i dispositivi della serie **enCoRe**™ II (**en**hanced **Co**mponent **Re**duction) dispongono di una memoria flash che consente la memorizzazione dei parametri dei sistemi wireless nonché la riprogrammabilità in-system. Tra le altre caratteristiche di rilievo da segnalare la ridotta dissipazione di potenza e il miglioramento delle potenzialità della CPU. I dispositivi della linea enCoRe USB dispongono di un oscillatore on chip, rendendo inutile il ricorso a un risonatore o a un quarzo esterni. La società ha inoltre integrato altri componenti esterni presenti nelle applicazioni USB low speed tra cui resistori di pull up, circuito di wake up e regolatore a 3,3 V.

238-04 BIVAROPTO™ PRESENTA NUOVI DISPLAY A SETTE SEGMENTI

La BivarOpto™, divisione Optoelettronica della Bivar, Inc., presenta una nuova e completa linea di display a sette segmenti che fornisce i massimi livelli di uniformità dell'industria, con processi di controllo BIN di massima precisione, per ottenere la massima coerenza di luminosità e colore. Questi nuovi display a sette segmenti, che impiegano gli stessi chip (piastrine) di LED utilizzati nei LED discreti e a montaggio superficiale (SMD) della BivarOpto, sono considerati ideali per le più esigenti applicazioni industriali, commerciali e medicali. La nuova serie BD è disponibile in una gamma di display LED a sette segmenti a una cifra, in sei altezze di cifra diverse, da 7 mm a 25,4 mm, in 24 stili e modelli diversi. Un circuito ad anodo comune o a catodo comune ne permette l'impiego in qualsiasi configurazione di display numerico, compresa una virgola a destra o a sinistra, e 18 caratteri alfanumerici. La gamma di lunghezze d'onda standard va da 465 nm a 630 nm, coprendo l'intera gamma dei colori standard (blu, verde, giallo e rosso). Il colore standard della faccia anteriore è nero con segmenti bianco-diffusi. Sono disponibili altri colori del segmento o della faccia. La BivarOpto propone anche display a segmento da 2 a 8 cifre con opzioni di display di superamento (+/-) e di clock.

238-05 FOX, UN COMPLETO SISTEMA LINUX SU SINGOLA SCHEDA

Il FOX è una scheda di sviluppo che utilizza il modulo ETRAX LX100 MCM 4+16 prodotto dalla Axis, leader mondiale nelle network camera e nei printer server. Il FOX contiene tutta la potenza di un sistema Linux completo in una scheda di piccole dimensioni (66x72 mm).
La scheda Fox è utilizzabile sia come dispositivo

di rete autonomo - per applicazioni quali micro web server, proxy server, router, ecc. - sia come modulo socket da innestare su una scheda madre per dare ai tuoi progetti tutta la potenza di Linux e le capacità di networking su rete TCP/IP con uno sforzo minimo.

Caratteristiche software:
- Linux kernel 2.6.
- WEB server, FTP server, SSH, SCP, TELNET e PPP preinstallati.
- Supporta ai convertitori USB/seriali basati su chip FTDI.
- Supporto alle chiavi di memoria USB.
- Ambiente di sviluppo (SDK) Open Source per sistemi Linux.
- Compilatore GNU C con interfaccia web disponibile gratuitamente su Internet per la compilazione di piccole applicazioni senza dover installare l'SDK.

Caratteristiche Hardware:
- CPU Axis ETRAX 100LX RISC 32 bit 100MHz, 100MIPS.
- 16MB di memoria RAM memory, 4MB di memoria FLASH.
- 1 porta Ethernet a 10/100 Mb/s.
- 2 porte USB 1.1.
- 1 porta seriale TTL a 3,3 volt per consolle.
- Due connettori di espansione per diverse interfacce.
- Alimentazione singola a 5 Volt (consuma 280mA senza dispositivi USB connessi).
- Dimensioni: 66x72x19mm - Peso: 37g.

Connettori di espansione:
- Fino a 2 porte parallele.
- Fino a 4 porte IDE.
- Fino a 2 porte SCSI o una porta SXSI Wide.
- Un bus I2C.
- Fino a 62 linee di I/O.
- Fino a 3 porte seriali.

238-06 MACH X: KIT DI PROGRAMMAZIONE E DEBUGGING PER I PICMICRO

Mach X è un programmatore completo, ottimo sia per i dispositivi Flash che OTP. La velocità di programmazione è garantita da una CPU PIC18 a 40 MHz. Sfruttando la logica ICSP è possibile programmare chip con vari packages, da 8 a 40 pin, tramite un unico zoccolo ZIF a 40-pin. Mach X supporta l'In-Circuit sia per la programmazione che per il debug.

Caratteristiche tecniche:
- Dispositivi supportati: PIC12, PIC14, PIC16, PIC18, dsPIC, MCPxxxx(CAN Bus chips).
- Programma diversi chip con package DIP (8-40pins) attraverso uno zoccolo ZIF a 40-pin.
- La programmazione In-Circuit funziona mediante un connettore standard ICD.
- Funziona come ICD debugger con il software CCS Windows IDE compiler.
- Interfaccia USB: nessun alimentatore richiesto (sono disponibili driver e software per Linux e Windows).
- Tensione di verifica impostabile dall'utente (2V to 5.5V), importante per soddisfare le specifiche Microchip per la produzione mediante ICSP.
- Pulsante Start/Repeat sul programmatore per una semplice programmazione in volumi.
- Numerazione avanzata dei dispositivi: il software prevede un efficiente sistema per la serializzazione dei componenti, sia in modalità binaria che in ASCII.
- Possibilità di salvare un file di log.
- Calcolo e programmazione della calibrazione per i dispositivi con l'oscillatore interno.

UN EMBEDDED LINUX A STATO SOLIDO

PRONTO PER SVILUPPARE CENTINAIA DI APPLICAZIONI

Dal blocco note di Fare Elettronica una raccolta di idee da tenere sempre a portata di mano.

Notepad

Questa rubrica ha lo scopo di fornire degli schemi applicativi o idee di progetto dei componenti elettronici più interessanti, selezionati per voi dalla redazione. Tutti gli schemi presentati sono elaborazioni di quelli ufficiali proposti dai produttori nella documentazione ufficiale.

ETHERNET FACILE

Collegare il proprio hardware alla rete ethernet non è mai stato così facile.
Non c'è bisogno di gestire lo stack TCP/IP, non c'è bisogno di prevedere un menu di configurazione, ma basta solo una porta seriale e a tutto il resto provvede NEMO10. In figura una applicazione con un microcontrollore.

SINTESI VOCALE AD ALTE PRESTAZIONI

I chip Sensory della serie SC-6x consentono la sintesi vocale garantendo una qualità elevata. In figura è riportata una applicazione del chip SC-691 che pilotato da un microcontrollore riproduce l'audio appositamente compresso e memorizzato nella memoria esterna.

+Vdd
EEPROM
Microcontroller
4x100k
100k
4.7k
4.7k
4.7k
1
Vdd
6 /INRDY
5 /OUTRDY
3 R/W
4 /STROBE
4
36..39 DATA(0..3)
35 DATA/COMMAND
12 /RESET
Vss
16
DACP 47
DACM 45
+
-
SP
Speaker

ACQUISIZIONE DATI MULTICANALE CON ADS1244/45

L'ADS1244/45 è un ottimo convertitore A/D a basso consumo ed alta precisione di Texas Instruments adatto all'acquisizione e conversione di un singolo canale analogico.
Tuttavia è possibile impiegare questi dispositivi per realizzare un sistema di acquisizione dati multicanale.
La figura che segue ne mostra il collegamento.

CHIAVE ELETTRONICA CON DALLAS I-BUTTON

Il sistema IBIS è una scheda di dimensioni ridottissime che permette di utilizzare immediatamente le chiavi DALLAS i-BUTTON del tipo DS1971.
Consente la lettura/programmazione di una master key e di un massimo di tre chiavi slave. L'inserzione di una chiave correttamente programmata provoca la chiusura del relè.
La memoria della scheda può essere resettata per programmare nuove chiavi.

FUSIBILI SOTTO CONTROLLO

Lo schema applicativo di figura mostra come controllare lo stato di due fusibili mediante un microcontrollore e determinare quale dei due sia interrotto. Il principio si basa sulla conversione AD, infatti se i due fusibili sono integri la tensione in ingresso è data dal partitore formato da R e dal parallelo tra R1 ed R2. Nel caso in cui uno dei due fusibili è interrotto, il rapporto di partizione cambia in quanto la rispettiva resistenza viene esclusa dal partitore. Se entrambi i fusibili sono interrotti, la tensione all'ingresso del convertitore AD è nulla grazie al pull-down costituito da R.
Un sistema di questo tipo è utile in tutte le apparecchiature che hanno un sistema di alimentazione ridondante: in questo modo è possibile risalire a quale dei due moduli di alimentazione è eventualmente non funzionante.

Elettronicando

Il controllo del flusso di cariche elettriche è il meccanismo che sta alla base di tutti i dispositivi elettronici. È necessario quindi poter disporre di cariche elettriche mobili, ed i processi che ne governano il movimento devono essere controllabili.
Discuteremo in questa lezione le caratteristiche dei semiconduttori, e come si differenziano dagli isolanti e dai conduttori.
L'uso di impurezze è fondamentale per la creazione della giunzione pn, che descriveremo in modo diretto e pratico.
Un datasheet ci sarà di aiuto per comprendere le caratteristiche di un diodo al Silicio, ed alcuni esempi saranno utili per chiarire il suo funzionamento.

LA CONDUZIONE NEI METALLI

L'elettrone è la principale particella carica negativamente, la cui carica è stata determinata come 1.60 x 10^{-19} C. Ogni coulomb di carica contiene quindi 6 x 10^{18} elettroni, il che significa che una corrente pari a un picoampere rappresenta un movimento di – approssimativamente – 6 milioni di elettroni.
Diamo per scontato che abbiate, almeno una volta, incontrato il *modello dell'atomo di Bohr*. Secondo *Niels Bohr* (fisico danese, 1885-1962) l'atomo è formato da un *nucleo* centrale attorno al quale ruotano, in orbite ellittiche denominate "*orbitali*", tutti gli elettroni. Questi orbitali, ai quali corrispondono le cosiddette "*bande energetiche*", di cui però non ci occupiamo, possiedono varie forme e sono situati ciascuno a distanze diverse dal nucleo. Questa teoria valse a Bohr il premio Nobel per la Fisica nel 1922.
Solitamente vengono chiamati *elettroni di valenza* quegli elettroni che, occupando l'orbitale più esterno, contengono quella carica negativa che rende neutra la configurazione dell'atomo.
In particolari condizioni uno o più elettroni di valenza possono essere rimossi, lasciando così una conformazione di ione positivo; analogamente, l'aggiunta di uno o più elettroni di valenza può generare uno ione negativo.
Facciamo un esempio. Il comune sale da cucina è cloruro di sodio, formato da uno ione cloruro (negativo) e da uno ione di sodio (positivo). Lo ione cloruro è ottenuto con l'aggiunta di un elettrone di valenza, mentre lo ione sodio è generato dalla rimozione di un elettrone di valenza.
La maggior parte dei metalli (Alluminio, Rame, Oro…) sono buoni conduttori di elettricità; la loro resistività è molto piccola ed è una funzione lineare della temperatura. Nei metalli gli elettroni di valenza sono così debolmente legati ai loro atomi che è sufficiente una piccola energia affinché questo legame si rompa, generando uno ione positivo ed un elettrone che diviene libero e disponibile alla conduzione (figura 1).
Poiché questo avviene già con l'energia termica disponibile alla temperatura ambiente, il

di Riccardo Nicoletti
r.nicoletti@farelettronica.com

I semiconduttori e il diodo pn

numero di elettroni liberi in un metallo è circa uguale a quello degli atomi del reticolo ed è quindi molto elevato. Un valore tipico di concentrazione di elettroni in un metallo è $n=10^{28}$ elettroni/m^3.

LA CONDUZIONE NEI SEMICONDUTTORI

Il Germanio ed il Silicio sono i due semiconduttori più importanti (senz'altro lo sono storicamente, data la recente comparsa di nuovi materiali) tra quelli impiegati nei dispositivi elettronici. La loro struttura cristallina è costituita da una ripetizione regolare di una cella tetraedrica con un atomo su ciascun vertice. La figura 2 mostra tale struttura riportata in due dimensioni.

L'atomo del Silicio è tetravalente, avendo quattro elettroni nel suo orbitale più esterno. Poiché l'atomo completa l'orbitale quando ne ha otto, esso si pone in una situazione ideale legandosi con quattro atomi attigui nel reticolo. Quindi ciascuno ione di Silicio condivide una coppia di elettroni con ciascuno dei quattro ioni vicini. Questa situazione prende il nome di "*legame covalente*". La situazione sembra favorevole, ma l'unico problema è che, per il fatto che i quattro elettroni di valenza sono adesso legati ad altri elettroni, il cristallo presenta una bassa conducibilità.
Ad una bassa temperatura (es. 0 K) il cristallo si comporta come un *isolante*, dato che non vi sono portatori di elettricità liberi di muoversi.

Alla temperatura ambiente alcuni dei legami covalenti si romperanno a causa dell'energia termica acquistata dal cristallo e pertanto degli elettroni verranno liberati. Questi elettroni "condivisi" mancanti nella struttura, daranno luogo a quelle che comunemente si chiamano "*lacune*" (figura 3). La cosa affascinante è che queste "*vacanze*" possono muoversi nel semiconduttore in maniera del tutto confrontabile con quella degli elettroni, dando quindi luogo ad un movimento di cariche elettriche libere positive. La quantità di carica associata ad una lacuna ha dunque lo stesso valore di quella di un elettrone, ma con segno positivo.

Figura 1 Rappresentazione schematica in due dimensioni del reticolo cristallino di un metallo; sono visibili gli elettroni di valenza disponibili per la conduzione (modello del "gas elettronico")

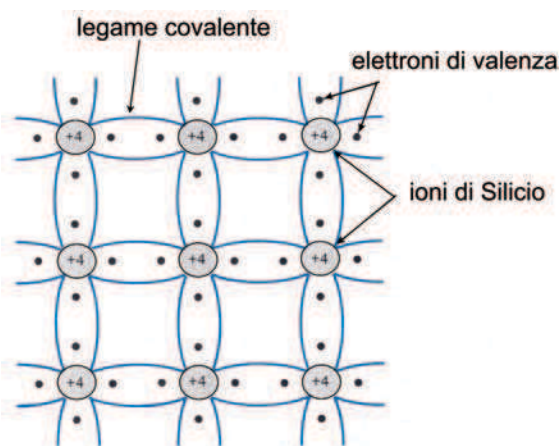



Figura 2 Struttura cristallina del Silicio illustrata in due dimensioni

Se una lacuna è generata dalla mancanza di un elettrone nel legame, è quindi chiaro che in un semiconduttore puro queste due quantità sono uguali. Il numero degli elettroni liberi è uguale a quello delle lacune. Si indicano con n e p, rispettivamente, la concentrazione degli elettroni liberi e delle lacune, per cui n=p; tale concentrazione viene detta "*concentrazione intrinseca*" del semiconduttore e si indica col pedice "i". Per avere un'idea, pensate che a 300 K (cioè 27°C) la concentrazione degli elettroni liberi nel Silicio vale $n_i=10^{10}$ cariche/cm^3; poiché la concentrazione degli atomi è 5 x 10^{22} atomi/cm^3, solo un atomo ogni 10^{12} circa è ionizzato alla temperatura ambiente.

IMPURITÀ: DONATORI E ACCETTORI

Se al semiconduttore intrinseco si aggiunge una piccola percentuale di atomi trivalenti o pentavalenti si ottiene un semiconduttore "*drogato*" (o "*impuro*", o "*estrinseco*"). Tra poco sarà chiara l'utilità di questa aggiunta.

Donatori – Se l'elemento "drogante" possiede cinque elettroni di valenza, si ottiene la situazione riportata in figura 4. Gli atomi droganti occuperanno dei posti nel reticolo cristallino e quattro dei cinque elettroni formeranno dei legami covalenti con atomi vicini; il quinto rimane libero e sarà disponibile per la conduzione.

Figura 3 Un legame covalente si spezza, dando origine ad un elettrone libero e ad una lacuna

L'energia necessaria per liberare questo elettrone dall'atomo è circa venti volte inferiore a quella necessaria a strapparlo nel caso di Silicio intrinseco.
Elementi pentavalenti utilizzati come droganti sono l'Antimonio, il Fosforo e l'Arsenico. Tali elementi sono detti "*impurità di tipo n*" o "*donatori*".

Accettori – Se al semiconduttore si aggiunge un'impurità trivalente (Boro, Gallio o Indio) solo tre legami covalenti possono essere formati,

Figura 4 Reticolo cristallino in cui un atomo di Silicio è sostituito da un'impurità di tipo n

Figura 5 Reticolo cristallino in cui un atomo di Silicio è sostituito da un'impurità di tipo p

mentre nel quarto sarà presente una lacuna. Si veda la figura 5. Gli elementi trivalenti generano quindi lacune, che sono in grado di accettare elettroni. Questi elementi sono detti "*accettori*" o "*impurità di tipo p*".

Quello che si ottiene con il drogaggio di un semiconduttore intrinseco è, in definiva, un aumento della conducibilità; in secondo luogo tale operazione permette di ottenere, a scelta, un conduttore in cui i portatori di carica elettrica sono in maggioranza elettroni o lacune. In un semiconduttore di tipo n gli elettroni sono i portatori di maggioranza, mentre in un semiconduttore di tipo p le lacune sono portatori di maggioranza e gli elettroni sono portatori di minoranza.

GIUNZIONE PN

La giunzione pn a circuito aperto

Consideriamo un cristallo di semiconduttore ed introduciamo da un lato impurezze di tipo p e dall'altro di tipo n. Si ottiene quella che si chiama "*giunzione pn*", o "*diodo a giunzione*", riportata schematicamente in figura 6. Vediamo di commentare la figura, spiegando cosa accade.
Gli elettroni presenti nella parte di destra (facenti parte degli atomi donatori) si accorgono della mancanza di elettroni che vi è nella parte dove c'è l'impurezza di tipo p e, per il fenomeno della diffusione, "*diffondono*" nella parte p *ricombinandosi* con le lacune presenti; l'effetto di questa "*migrazione*" è che a destra non abbiamo più elettroni, mentre a sinistra questi si sono ricombinati con le lacune. Ad una certa distanza dalla giunzione non abbiamo quindi a disposizione cariche libere; questa zona prende il nome di *zona di svuotamento* (o regione di *carica spaziale*) ed è popolata da cariche scoperte. Lo ione donatore è disegnato con un segno "+" in quanto, dopo che ha donato un elettrone, esso diventa uno ione positivo; lo ione accettore è indicato con un segno "-" poiché esso diviene uno ione negativo dopo aver accettato un elettrone.

La giunzione pn come dispositivo raddrizzante (diodo)

La caratteristica elettrica fondamentale della giunzione pn sta nel fatto che essa costituisce un elemento *raddrizzante*, capace cioè di facilitare il passaggio della corrente in un verso e di ostacolarlo nell'altro. Vediamo, in un modo del tutto qualitativo, come questo possa accadere.
Il simbolo del componente è costituito da un triangolo nero (zona p, anodo, A) la cui punta indica il verso di scorrimento della corrente, e da una sbarretta trasversale (zona n, catodo, K).

Polarizzazione inversa – Supponiamo di collegare il morsetto negativo di una batteria alla regione p ed il morsetto positivo alla zona n (figura 8). Le lacune presenti nella parte p vengono richiamate dal morsetto negativo e si allontanano pertanto dalla giunzione; analogamente gli elettroni nella parte n si allontanano dalla giunzione richiamati dal potenziale positivo del morsetto della batteria. Quello che si ottiene è un allargamento della regione di svuotamento. In questa situazione non si creano cariche mobili e la corrente che circola è, nominalmente, zero. In realtà una piccola corrente circola, uni-

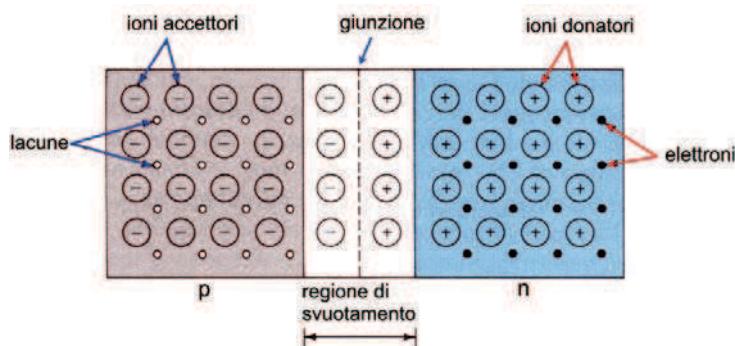



Figura 6 Rappresentazione schematica di una giunzione pn

Figura 7 Simbolo grafico del diodo

camente dovuta alla generazione termica di coppie elettrone-lacuna ed essa viene detta "*corrente di saturazione inversa*" I_0.
Quindi: anodo negativo rispetto al catodo → polarizzazione inversa → *interdizione* (cioè la corrente non passa).

Attenzione: in questa situazione si crea alla giunzione una distribuzione di cariche affacciate che ricorda molto un condensatore. La capacità del diodo in polarizzazione inversa viene molto usata nei moderni dispositivi.
Se si considera che essa può cambiare al variare della tensione di polarizzazione inversa applicata, si crea così un condensatore variabile elettricamente (*diodo varicap*).

Polarizzazione diretta – Quando la polarità

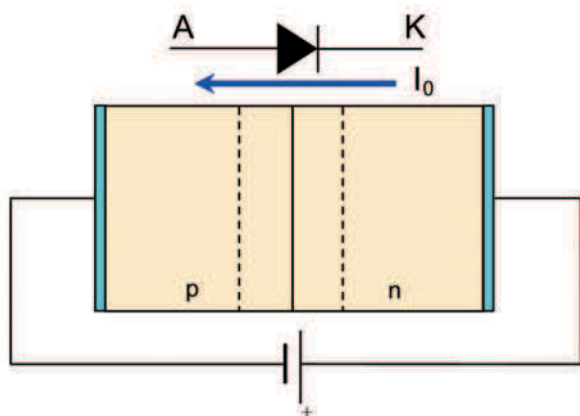



Figura 8 Diodo in polarizzazione inversa

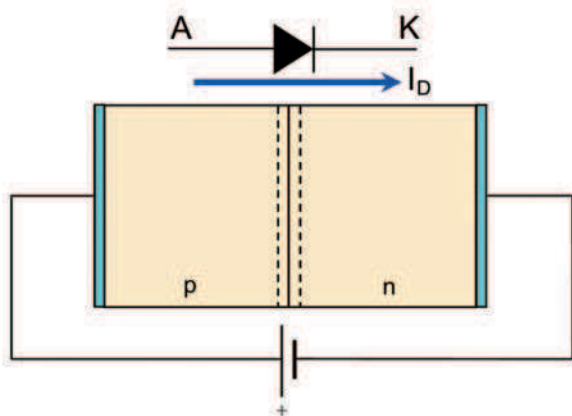



Figura 9 Diodo in polarizzazione diretta

della batteria è col morsetto "+" rivolto dalla parte "p" della giunzione, le lacune tendono ad essere iniettate verso la giunzione e da qui a diffondere nella regione n.
Gli elettroni sono spinti dal morsetto negativo a muoversi verso la giunzione e da qui diffondono nella regione *p*. poiché un elettrone che si muove verso sinistra equivale ad una lacuna che si muove verso destra, le due componenti di corrente si sommano, generando quindi una corrente di polarizzazione diretta I_D.
Quindi: anodo positivo rispetto al catodo → polarizzazione diretta → conduzione di corrente.

La caratteristica tensione-corrente del diodo

La corrente del diodo fluisce dall'anodo al catodo in polarizzazione diretta, ed aumenta esponenzialmente al crescere della tensione V_D applicata al diodo (figura 10).
Per completezza riportiamo qui l'espressione della corrente nel diodo in funzione della tensione V_D:

$$I_D = I_O \left(e^{\frac{V_D}{\eta V_T}} -1 \right)$$

dove

- I_0 è la corrente di saturazione inversa (a volte indicata con I_R , I_L o I_S).
- η è una costante (si pronuncia "eta") che per il Silicio è compresa tra 1 e 2.

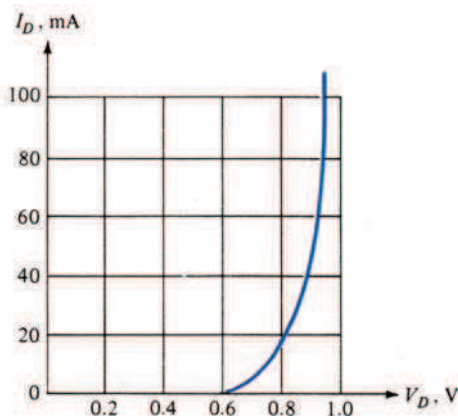



Figura 10 Caratteristica tensione-corrente in polarizzazione diretta, per un diodo al Silicio

- V_T si chiama "equivalente in Volt della temperatura" e rappresenta l'energia termica degli elettroni alla temperatura di lavoro (a temperatura ambiente V_T vale circa 26 mV).

Quando V_D è tale da rendere l'anodo positivo rispetto al catodo, finché V_D è inferiore a 0.5 V non accade nulla di rilevante, ma appena V_D supera 0.6 V la corrente che attraversa il diodo aumenta esponenzialmente.
Il valore di tensione oltre il quale la corrente aumenta esponenzialmente si chiama *tensione di soglia* e si indica con $V\gamma$ (per i diodi al Germanio $V\gamma$ è circa 0.2 V).

Quando il diodo è polarizzato inversamente, cioè V_D è negativa, la corrente vale circa I_0 e questa situazione di lavoro si chiama appunto "*saturazione inversa*".

La corrente che attraversa il diodo è piccolissima, dell'ordine dei nanoampere alla temperatura ambiente.

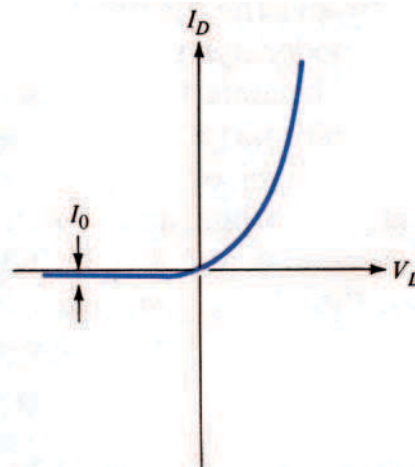



Figura 11 Caratteristica tensione-corrente, dove è posta in evidenza la corrente di saturazione inversa

Il diodo come raddrizzatore

Abbiamo quindi imparato che il diodo è un componente "*raddrizzante*". Questo termine deriva dal fatto che, se alimentiamo il diodo con un segnale sinusoidale, esso si pone in conduzione solo per quella parte del segnale che rende positivo l'anodo rispetto al catodo. Questo si verificherà solo in corrispondenza delle semionde positive, pertanto le semionde negative (che invece rendono l'anodo negativo rispetto al catodo) non pongono il dispositivo in conduzione e quindi non lo attraversano. A valle del diodo si trova quindi un segnale che presenta solo la sua parte "alta", avendo perduto la parte inferiore. Da qui si dice che il segnale è stato "raddrizzato".

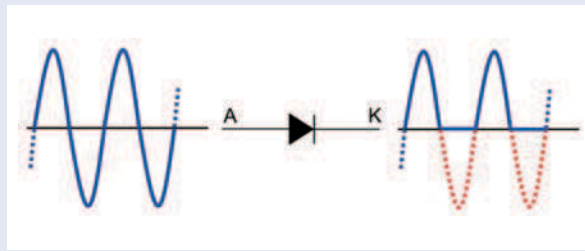



Figura 13 *Il diodo elimina le semionde negative e "raddrizza" il segnale sinusoidale*

La caratteristica tensione corrente del diodo presenta un andamento curioso nella parte di polarizzazione inversa, cioè quando V_D diviene negativa. All'aumentare del modulo di V_D la corrente che attraversa il diodo si mantiene di valore costante I_0 fino alla regione di "*ginocchio*" in cui la caratteristica comincia a "*piegare*" (figura 12). In tale regione la corrente tende ad aumentare molto per piccole variazioni di V_D. In tale regione si è soliti indicare la tensione ai capi del diodo con V_R (cioè "Reverse Voltage") per capire che il diodo sta lavorando in polarizzazione inversa. Superato il ginocchio, ad impercettibili aumenti di V_R corrispondono grandi aumenti di I_R. Tale regione prende il nome di "*breakdown*" e la tensione V_R, praticamente costante, viene indicata con V_Z. Il fenomeno fisico che regola questo comportamento prende il nome di *moltiplicazione a valanga* ed è sostanzialmente un moltiplicarsi dei portatori all'interno della giunzione per effetto di urti tra coppie lacuna-elettrone e ioni del reticolo. Perdonate la sintesi, ma vi sono ottimi testi dove approfondire l'analisi di tale (complesso) fenomeno.

Figura 12 *Caratteristica completa di un diodo al Silicio*

Un esempio di caratteristica tensione-corrente per un generico diodo al Silicio è riportata in figura 12; si noti la corrente di saturazione inversa I_0 nella zona di polarizzazione inversa, dove la scala del disegno è stata esagerata per motivi di chiarezza.

Caratteristica logaritmica

Spesso nei datasheet dei diodi viene riportata la caratteristica tensione-corrente logaritmica (figura 14). Come si ricava? Non vi spaventate! Se effettuate il logaritmo dei due membri dell'espressione

$$I_D = I_O \left(e^{\frac{V_D}{\eta V_T}} - 1 \right)$$

ottenete

$$Log\, I_D = Log \left[I_O \left(e^{\frac{V_D}{\eta V_T}} - 1 \right) \right]$$

supponiamo che $V_D >> V_T$ (ipotesi del tutto ragionevole, dato che $V_T \approx 25$ mV alla temperatua ambiente) in modo che sia $e^{\frac{V_D}{\eta V_T}} >> 1$, quindi possiamo trascurare l'unità nella parentesi

$$Log\ I_D = Log\left(I_O\ e^{\frac{V_D}{\eta V_T}}\right)$$

da cui, essendo il logaritmo del prodotto la somma dei logaritmi dei singoli fattori, si ha

$$Log\ I_D = Log I_O + Log\ e^{\frac{V_D}{\eta V_T}}$$

da cui

$$Log\ I_D = Log I_O + 0{,}434\ \frac{V_D}{\eta V_T}$$

(perdonate la parentesi con un pò di matematica…)

La funzione $Log\ I_D = Log I_O + 0{,}434\ \frac{V_D}{\eta V_T}$

è rappresentata da una retta in un diagramma logaritmico. Un esempio è riportato in figura 14, dove si nota che per alte correnti le rette si incurvano per motivi che qui tralasciamo. Si noti che l'asse logaritmico delle ordinate con-

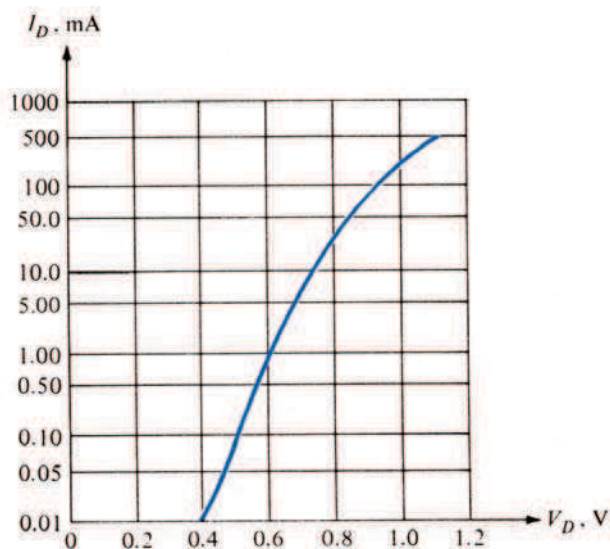



Figura 14 *Caratteristica tensione corrente con asse delle correnti logaritmico*

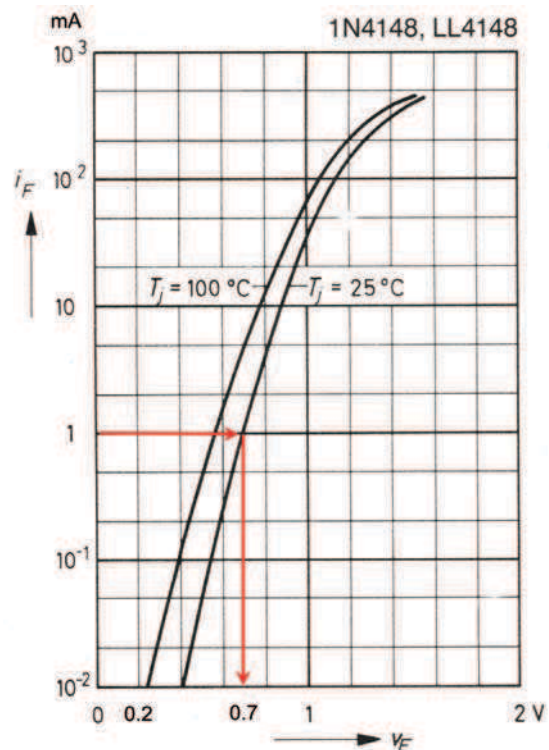



Figura 15 *Caratteristica VF-IF (zona di polarizzazione diretta) del diodo 1N4148*

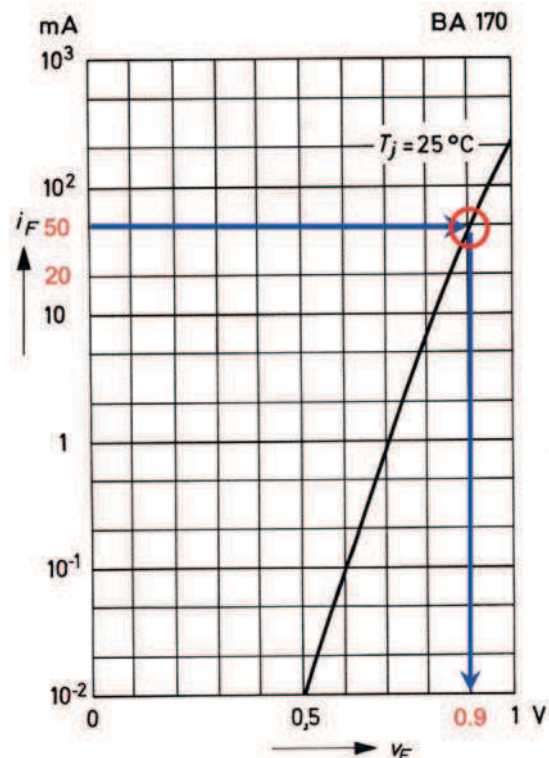



Figura 16 *Caratteristica VF-IF (zona di polarizzazione diretta) del diodo BA170*

sente di rappresentare correnti in rapporto fino a 5000 (si parte da 0.01 e si arriva a 500!).

Riportiamo due esempi di caratteristiche V-I per due diodi commerciali di tipo 1N4148 (per commutazioni veloci) e BA170 (per applicazioni generiche). I datasheet dei due diodi sono disponibili nel sito della Rivista per fare pratica con i parametri elencati che verranno discussi in seguito.

Interpretiamo la curva del diodo 1N4148. L'asse delle correnti (I_F) è tarato in milliampére, quello delle tensioni in volt. Sull'asse delle correnti sono riportati i valori 1-2-5-10 con loro multipli e sottomultipli, mentre le divisioni sull'asse delle tensioni segnano 1-2-4-6-8-10. Quindi, per esempio, alla temperatura di giunzione T_j=25 °C (curva situata più a destra), ad una corrente I_F=1mA corrisponde una tensione diretta ai capi del diodo pari a V_F=0.7 V (figura 15).

Vediamo un altro esempio. Per il BA170, ad una corrente I_F=50 mA corrisponde una tensione di polarizzazione diretta pari a V_F=0.9 V (figura 16).

Il datasheet, che vi invitiamo a leggere, contiene altre curve interessanti, che qui non stiamo ad esaminare.
Per la vostra curiosità, potete trovare come varia la massima potenza dissipabile in fuzione della temperatura ambiente, l'andamento della capacità di giunzione al variare della tensione inversa e la corrente di saturazione in funzione della temperatura di giunzione.

Figura 17 Un comune diodo al Silicio

Il componente

Il diodo a giunzione si presenta come un piccolo cilindretto di vetro o plastica su cui è impressa una fascetta color argento (o nera, o bianca), che indica il catodo (figura 17). In alcuni diodi con involucro di vetro la fascetta è di colore nero o rosso.
Nonostante la maggior parte del mercato sia occupata dai diodi al Silicio, anche il Germanio è utilizzato per costruire diodi. Il Silicio ha avuto un grande successo nella costruzione di dispositivi planari per alcune sue caratteristiche che lo rendono unico; ricordiamo che si lavora bene, è un elemento molto stabile ed a contatto con l'aria forma l'ossido SiO_2 che è l'isolante elettrico ideale per i circuiti integrati.
I diodi al Germanio, costruiti saldando dei sottili terminali in Oro ad una giunzione pn realizzata con Germanio, possiedono una piccola capacità di giunzione (a causa dell'utilizzo dell'Oro), brevi periodi di commutazione ed ottime caratteristiche alle radio frequenze; sono principalmente usati in circuiti logici e in demodulatori RF. Sono stati molto usati in passato per la loro bassa tensione di soglia, pari a circa 0.2 V.
I diodi al Silicio, dal canto loro, possiedono piccole correnti di perdita, alte tensioni inverse di breakdown e temperature massime di funzionamente fino a 200 °C, che consentono un'alta dissipazione di potenza anche in contenitori di dimensioni estremamente contenute. Queste loro ottime caratteristiche di dissipazione di potenza, unitamente alla maggiore semplicità costruttiva e all'avvento di nuovi dispositivi, hanno determinato nel tempo il minore utilizzo dei loro fratelli "minori" al Germanio.

COME SCEGLIERE UN DIODO

I diodi raddrizzatori vengono utilizzati per una vasta gamma di applicazioni; a seconda delle tecniche di costruzione, del drogaggio del semiconduttore e delle dimensioni del chip, si possono ottenere diodi capaci di sopportare una corrente massima che varia da 1 A a decine e centinaia di ampère, con tensioni di lavoro massime che vanno da qualche decina a varie centinaia di volt.
Ciascun Costruttore assegna le proprie sigle ai

diodi, generalmente contraddistinti da nomi del tipo 1N.XXXX, BAY.XXX o simili.
Quando si sceglie un diodo sono molti i parametri riportati nel datasheet che è bene considerare in fase di progetto.
Apriamo un datasheet di un diodo per applicazioni generiche (general purpose) e vediamo di capirci qualcosa.
Riportiamo di seguito alcuni parametri fondamentali, esplicitando le loro denominazioni in lingua inglese. Attenzione che sul datasheet sono generalmente riportati i loro *valori massimi ammissibili*.

Tensione diretta V_D (in inglese "*Forward voltage*" V_F) è la caduta di tensione ai capi del diodo dovuta alla corrente I_F che scorre in polarizzazione diretta.
Viene esplicitata quando scorre nel diodo un preciso valore di corrente. Ad esempio: V_F= 1V (max) quando I_F=80 mA.

Tensione inversa, V_Z (in inglese "*Reverse voltage*", V_R) è la massima tensione sopportabile in polarizzazione inversa. Esempio: per il diodo 1N4148 si ha V_R=75 V. Una precisazione: nel testo, per evitare confusione, abbiamo chiamato V_R tutti i valori di tensione applicati ai capi del diodo in polarizzazione inversa, e non il valore massimo ammissibile.
Questo valore massimo appare in figura 12 indicato come V_Z. Speriamo che questa indicazione non abbia generato confusione. Quando leggete il datasheet, il valore V_R è un valore limite.

Corrente diretta (*Rated forward current*) I_F in c.c. è la massima corrente che può scorrere in condizioni statiche, ad una certa temperatura. In condizioni dinamiche viene espressa come valor medio. Il suoi valore massimo è legato alla potenza dissipabile dal chip.

Potenza dissipabile P_{tot} è la potenza massima dissipabile da quel diodo alla temperatura specificata (di solito quella ambiente). È un parametro che dipende dalle dimensioni del

chip, dal contenitore e quindi dalla sua capacità di dissipare all'esterno il calore generato. Come esempio riportiamo il valore del diodo 1N4148, pari a P_{tot}=500 mW con T_{amb}=25 °C.

Reverse (Leakage) current I_R, è la massima corrente che può scorrere in polarizzazione inversa, con una specifica tensione inversa V_R. Per esempio, per il solito diodo 1N4148, si ha I_R=25nA se V_R=20 V.

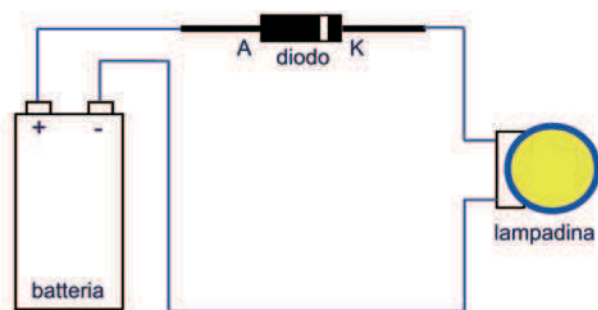
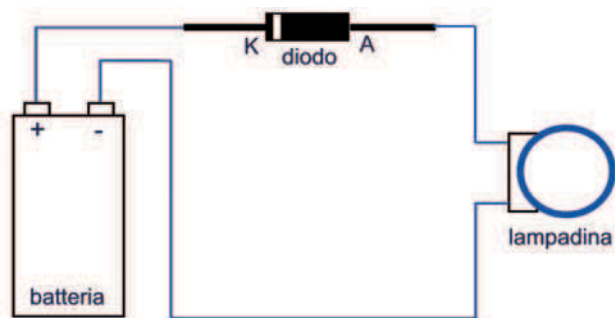
Reverse recovery time t_{rr} è un parametro che riguarda il funzionamento in commutazione. Sebbene non si siano affrontate le problematiche relative al funzionamento in condizioni dinamiche, ci preme spendere due parole su questo parametro.
Supponiamo di utilizzare un diodo in condizioni dinamiche, ad esempio applicando ai suoi capi una tensione che cambia polarità con una certa frequenza. Per un certo tempo il diodo sarà quindi polarizzato direttamente,

Figura 18 *Il diodo conduce quando l'anodo è rivolto verso il polo positivo della batteria e la lampadina si accende*

Figura 19 *Il diodo non conduce quando il catodo è rivolto verso il polo positivo della batteria; la lampadina in questo caso non si accende*

mentre per un altro periodo esso sarà polarizzato inversamente.
Sulla base di quanto abbiamo spiegato, in condizioni di polarizzazione inversa, la regione di svuotamente del diodo è priva di cariche libere; siamo quindi in questa situazione quando il segnale applicato al diodo lo polarizza inversamente.
La polarità del segnale poi cambia e polarizza direttamente il diodo; questo siginifca che il diodo ha bisogno di cariche mobili per riempire la regione di svuotamento e poi per entrare in conduzione.
Non può far questo in un tempo nullo, occorre cioè un certo intervallo di tempo affinché le cariche ripopolino la regione di svuotamento e si rendano così disponibili per la conduzione.
In fondo, abbiamo detto che il diodo in polarizzazione inversa è assimilabile ad una piccola capacità, quindi non ci stupisce il fatto che sia necessario del tempo affinchè questa "capacità" si carichi e si scarichi al variare della tensione ai suoi capi…. Questo parametro viene indicato specificando i valori di corrente diretta ed inversa, tra i quali viene eseguita la transizione.
A titolo di esempio, per i due diodi presentati in precedenza si ha:
per 1N4148 il datasheet riporta t_{rr} = 4 ns (max), mentre per il BA170 si ha t_{rr}=100 ns (max). Questo confronto fa capire quale dei due componenti è più indicato per applicazioni dove siano richieste veloci commutazioni.

UN SEMPLICE ESERCIZIO

Procuratevi una lampadina da 12V ed una pila con un valore di tensione comune, per esempio a 9V. Cercate un diodo avente una corrente diretta massima di un ampére, tipo 1N4001, 1N4002 o simili.
Questi diodi sono classificati proprio come "raddrizzatori", dispondendo di una corrente nominale di 1 A ed una tensione inversa massima fino a 1600 V (per il tipo EM513).
Collegate in serie i tre elementi. Vi accorgerete che la lampadina si accenderà solo quando l'anodo del diodo sarà collegato verso il polo

positivo della batteria (figura 18), mentre la lampadina non si accenderà se invertite il verso del diodo (figura 19). Quindi si ha conduzione solo quando l'anodo del diodo è rivolto verso un potenziale positivo capace di mandarlo in conduzione.
Se provate a collegare direttamente la lampadina alla pila (eliminando il diodo dalla serie dei tre), vedrete che in questo caso la luminosità sarà maggiore.
Infatti ai capi del diodo "cadono" 0.7 V dovuti alla conduzione; pertanto ai capi della lampadina non troviamo i 9 V della batteria, ma 9 V – 0.7 V = 8.3 V.

SIMULIAMO UN CIRCUITO RADDRIZZATORE

Simuliamo un semplice circuito raddrizzatore, aiutandoci con Cadlogix. Il nostro obiettivo è di verificare quanto abbiamo detto finora.
Apriamo Cadlogix e selezioniamo un NUOVO SCHEMA (figura 20).

Selezioniamo dal menu in alto SIMBOLI e poi LIBRERIA DI SIMBOLI. Si aprirà una finestra "Librerie" nella quale si deve scegliere "APRI".
Selezionare la libreria *simdevices.dsf* (figura 21) e cliccare APRI.

All'interno delle varie categorie selezionare i componenti necessari per comporre il nostro circuito, che al termine risulterà come in figura 22.

Il generatore scelto è sinusoidale; cliccando col pulsante destro e scegliendo MODELLO si possono impostare i parametri del generatore (figura 23).

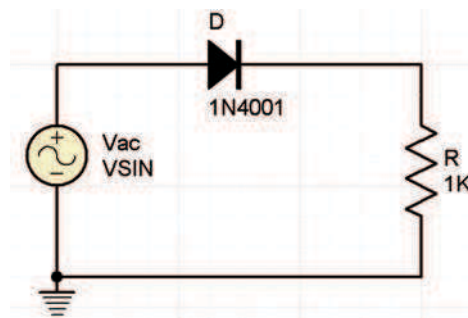



Figura 22 *Il semplice circuito raddrizzatore da simulare con Cadlogix*

Si noti che è stata scelta un'ampiezza (indicata con AC) pari a V_{ac}=10 V ed una frequenza f=100 Hz.

Domanda: con una frequenza pari a 100 Hz sapreste dire quanto vale il periodo corrispondente ? Poiché la frequenza è l'inverso del periodo si ottiene facilmente T= 1/100 = 0.01 s = 10 ms.

Il diodo è un semplice 1N4001, mentre la resistenza è stata scelta di valore abbastanza sostenuto per non far scorrere troppa corrente attraverso il diodo.

Selezioniamo CONFIGURA ANALISI dal menu ANALOGSIM; alla voce ANALISI IN TRANSITORIO clicchiamo su SETUP. Impostiamo l'intervallo di analisi che ci interessa (figura 24).

Mettiamo 100 ms nella voce TEMPO FINALE; essendo il periodo di un'onda pari a 10 ms, il valore 100 ms è più che ragionevole per poter visualizzare un certo numero di periodi.
L'intervallo "1 u" significa "1 microsecondo" e rappresenta l'intervallo tra due punti successivi

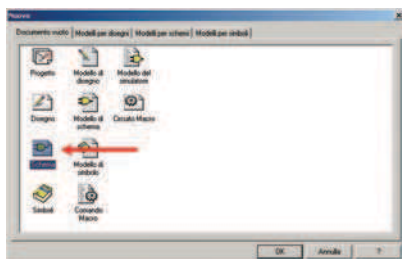



Figura 20 *Finestra dove selezionare "nuovo schema"*

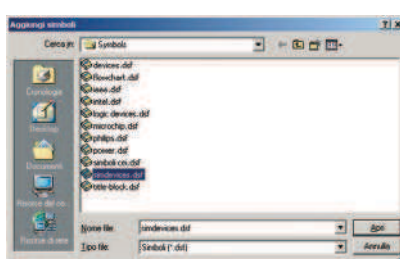



Figura 21 *Le librerie disponibili in cadlogix; selezionare "simdevices"*

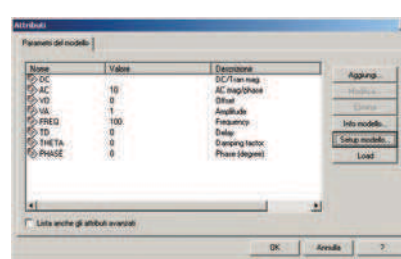



Figura 23 *I parametri del generatore sinusoidale*

analizzati dal simulatore; un intervallo piccolo consente di avere analisi accurate.

Clicchiamo su OK e poi spuntiamo la voce ANALISI IN TRANSITORIO nella finestra "CONFIGURA ANALISI" e clicchiamo OK per uscire (figura 25).

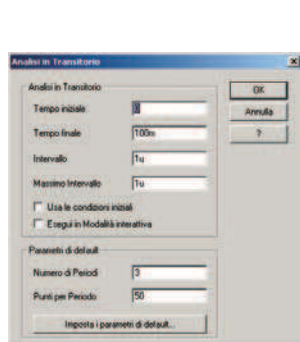



Figura 24 *Impostiamo i parametri di analisi*

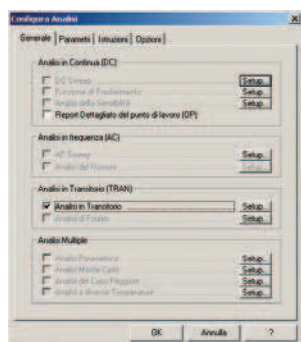



Figura 25 *Finestra per configurare l'analisi in transitorio*

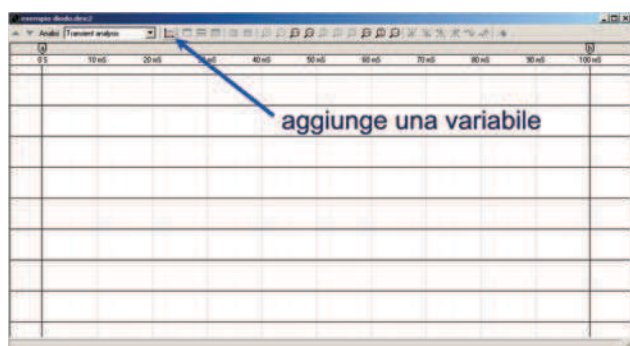



Figura 26 *Finestra dove verranno rappresentati i risultati dell'analisi*

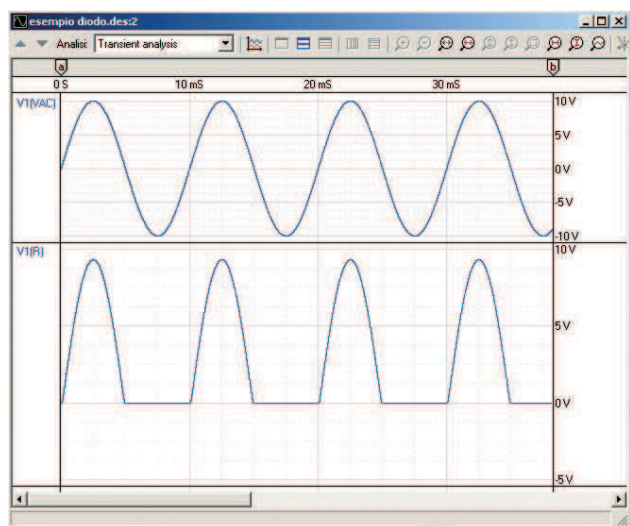



Figura 27 *I risultati dell'analisi, ottenuti con Cadlogix*

Dal menu ANALOGSIM clicchiamo RUN per avviare la simulazione. Si aprirà un diagramma inizialmente vuoto, nel quale occorre aggiungere le variabili da rappresentare.
Clicchiamo sul tasto indicato in figura per selezionare le variabili da aggiungere (figura 26).

Dopo aver cliccato il tasto indicato si selezionano le variabili di interesse, nel nostro caso V1(VAC) e V1(R). I grafici temporali delle variabili selezionate saranno riportati sul diagramma, come mostrato in figura 27.

Il file per eseguire la simulazione è disponibile nel sito della Rivista.

PROSSIMA PUNTATA

Nella prossima puntata vedremo alcune semplici applicazioni del diodo pn (rettificatore) ed introdurremo altri tipi di diodi, in particolare il diodo Zener e il diodo LED. Nell'ambito della puntata vedremo le indicazioni di massima per il progetto di un regolatore di tensione con diodo Zener.

BIBLIOGRAFIA

Alcuni paragrafi riguardanti la conduzione nei metalli e nei semiconduttori sono contenuti in:

1. E. Borchi, R. Nicoletti – "Elettromagnetismo", Vol. I "Elettricità", Ed. Esculapio, 2001, Bologna.

Fondamenti sui semiconduttori e capitoli sulla giunzione pn si trovano in:

2. J. Millman, C. C. Halkias – "Microelettronica", Ed. Boringhieri, 1978, Torino.
3. M. Gasparini, D. Mirri – "Dispositivi e circuiti elettronici", Vol. I, III ed., Ed. Calderini, Bologna, 1983.

Per i datasheet dei diodi:

4. Semiconductors ITT – "Diodes, Zener Diodes, Rectifiers", Manual 1985.
5. http://www.semiconductors.philips.com

Prima parte
Introduzione

Seconda parte
Il modulo di comando

Terza parte
Il booster, il modulo di servizio e il software di controllo

Il sistema DCC

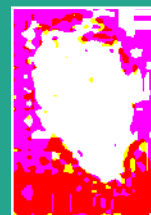
Dopo la descrizione del protocollo e del modulo di comando, in quest'ultima parte vedremo i moduli di potenza e l'utilizzo del software di controllo.

Come anticipato nei numeri precedenti il booster combina il segnale DCC proveniente dal modulo di comando con il segnale di potenza fornito dall'alimentatore. Un booster efficente deve inoltre consentire il controllo della potenza fornita al sistema per evitare corto circuiti o sovraccarichi che potrebbero danneggiare locomotive e accessori. Per raggiungere lo scopo in maniera ottimale, ho scelto un circuito integrato (L298) sviluppato per il controllo dei motori in DC. Ho inoltre inserito una sezione per il controllo dei sovraccarichi e per la ricezione del segnale di acknowledge durante le operazioni di programmazione.
A completamento del modulo booster c'è il modulo per la programmazione (RELBIN).
Si tratta di un modulo che sotto controllo della centralina invia l'uscita del booster al binario operativo o a quello di servizio. Questo modulo non è strettamente necessario in quanto con le dovute precauzioni è possibile programmare una locomotiva senza toglierla dal binario operativo.

DESCRIZIONE ELETTRICA E FUNZIONALE DEI MODULI

Modulo HBOOST

Lo schema elettrico del booster (HBOOST), mostrato in figura 1, può essere diviso in tre sezioni. In alto a destra la sezione di alimentazione costituita dal classico 7805 che preleva la tensione di alimentazione del modulo (connettore X2) e la riduce a 5 volt. La tensione così ottenuta viene utilizzata per alimentare le parti logiche del modulo e tramite il pin 4 di J1 viene inviata all'esterno per alimentare il modulo ENCSER. Un LED connesso al connettore J2 indica la presenza del segnale di alimentazione.
La sezione di potenza è costituita dall'integrato L298 (IC2). È un doppio driver di potenza in grado di controllare due motori DC oppure un motore passo - passo a 4 fasi con correnti di uscita fino a 2 Ampere per ciascuna sezione. Nel nostro caso le due sezioni del L298 sono connesse in parallelo per far si che venga fornita la massima corrente disponibile. Sulle uscite dei driver sono posti dei diodi di protezione con lo scopo di evitare extra tensioni che potrebbero essere generate da carichi ti tipo induttivo presenti sul plastico. I pins di SENSE (SEN_A e SEN_B) sono connessi a massa attraverso due resistori di potenza di basso valore ohmico (R4 e R5). Da questi pin viene prelevata una tensione proporzionale alla corrente di uscita. Questa tensione viene ripulita tramite un filtro passa basso (C2, R3) e inviata alla sezione costituita dai comparatori (IC1A e IC1B) che si occupano di controllare i livelli della corrente in uscita. Il comparatore IC1A, connesso in configurazione invertente, ha come tensione di riferimento 1.02V. La sua uscita va agli ingressi di ENABLE (Pin 6 e 11) di IC2 e tramite il connettore J1 (Pin 5) raggiunge il modulo ENCSER. L'uscita del comparatore va a 0 quando la corrente in uscita dal HBOOST supera i 4 ampere. In questo caso IC2 viene disabilitato e l'errore notificato al firmware di controllo (residente su ENCSER). La connessione diretta a IC2 garantisce una disattivazione immediata delle uscite del booster.
Il comparatore IC1B ha come tensione di riferi-

di Corrado D'Achille
wizgate@tiscali.it

Il booster, il modulo di servizio e il software di controllo

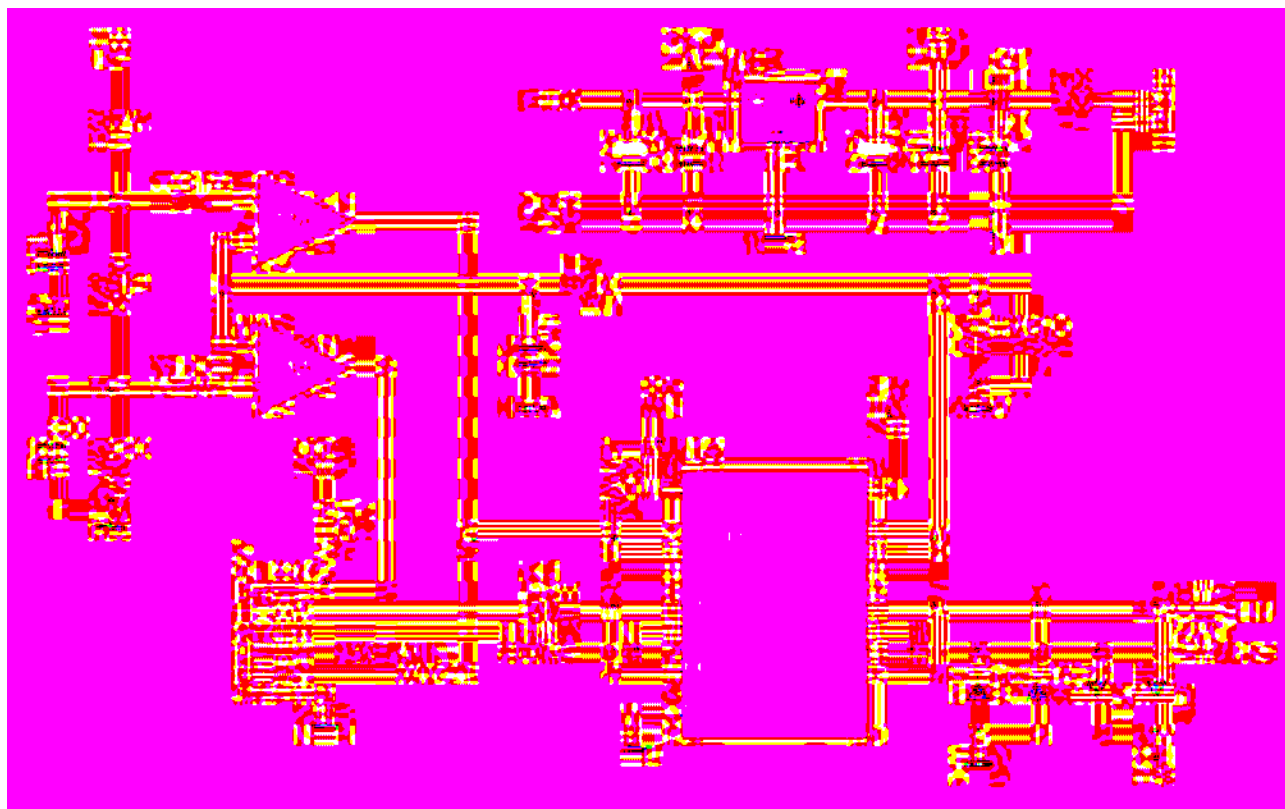



Figura 1 Schema elettrico del modulo HBOOST

mento 18mV e scatta quando la corrente di uscita supera i 60mA. Questo segnale viene inviato a ENCSER per notificare che l'operazione di programmazione è andata a buon fine.
Il modulo viene pilotato attraverso i pin 2 e 3 del connettore J1 (segnali DCCL e DCCH). I segnali sono in opposizione di fase.

Modulo RELBIN

Il modulo RELBIN, il cui schema è mostrato in figura 2, si occupa del controllo del binario. Il suo scopo è di inviare il segnale DCC (solo il segnale positivo) o al binario operativo o a quello di programmazione. Per raggiungere lo scopo, il circuito utilizza un relè ad uno scambio, controllato da un segnale proveniente dal modu-

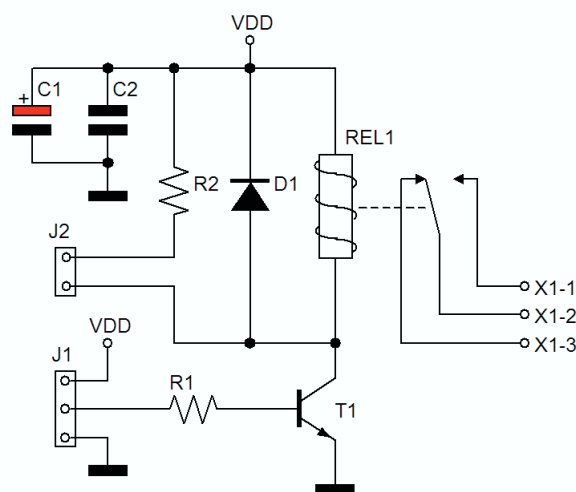



Figura 2 Schema elettrico del modulo RELBIN

Elenco componenti del modulo HBOOST	
Sigla	**Valore**
R1	39,7 KΩ 1/4 W 1%
R2	10 KΩ 1/4 W 1%
R3, R7, R8	10 KΩ 1/4 W
R4,R5	0,56 Ω 4W
R6	180 Ω 1/4W 1%
R9, R10	3,3 KΩ 1/4W
R11	1,8 KΩ 1/4W
R12	470Ω 1/4W
C1, C3, C5, C7,C8	100 nF 63 V ceramico
C2	10 nF 63 V ceramico
C4	2200 mF 25 V elettrolitico
C6	100 mF 16 V elettrolitico
D1,D2, D3,D4	BY251
DL1	Diodo LED
IC1	LM393N
IC2	L298
IC3	7805T
J1	Connettore da C.S. passo 2.54 6 poli
J2	Connettore da C.S. passo 2.54 2 poli
X1,X2	Morsettiera da C.S. passo 5.08 2 poli

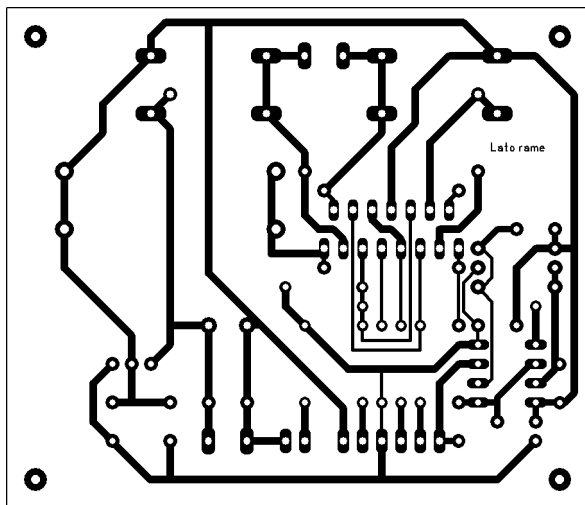



Figura 3 Circuito stampato in scala 1:1 (lato rame) del modulo HBOOST

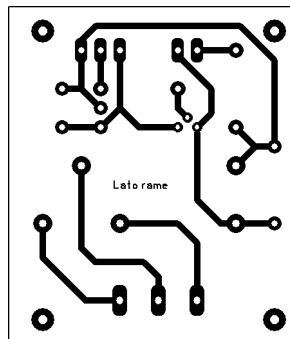



Figura 4 Circuito stampato in scala 1:1 (lato rame) del modulo RELBIN

lo ENCSER.
Quando il relè è a riposo, il segnale DCC viene inviato al binario operativo.
Viceversa durante la fase di servizio il relè viene attivato è il segnale DCC viene inviato al binario di servizio.
Il circuito di controllo del relè è composto da un transistor (T1) in configurazione ad emettitore comune. La base di T1 viene pilotata dal modulo ENCSER (Segnale SERV).
La resistenza R1 limita la corrente di base. Il collettore di T1 pilota il relè. In parallelo al relè è posto un diodo (D1) che taglia le extra-tensioni prodotte dalla bobina a seguito di un'attivazione. Un Led di segnalazione, connesso al connettore J2, segnala quando il relè è attivo (e quindi il sistema è in modalità di servizio).

IL MONTAGGIO

Nelle figure 3, 4, 5 e 6 trovate i disegni dei circuiti stampati e dei piani di montaggio dei moduli. Nel modulo HBOOST prestate attenzione ai ponticelli montati sotto a IC2 e vicino

Elenco componenti del modulo RELBIN	
Sigla	**Valore**
R1	2.2 KΩ 1/4 W
R2	390Ω 1/4 W
C1	100mF 10V Elettrolitico
C2	100nF 63V Ceramico
D1	1N4004
T1	BC337
DL1	Diodo Led
REL1	Relè 6 volt 1 scambio 4 ampere (Finder 40.51)
J1	Connettore da C.S. passo 2.54 3 poli
J2	Connettore da C.S. passo 2.54 2 poli
X1	Morsettiere da C.S. passo 5.08 3 poli

Figura 5 *Piano di montaggio del modulo HBOOST*

a diodi di protezione, usate del filo isolato al fine di evitare dei corto circuiti. Per gli altri componenti valgono le solite regole, ossia procedete nel montaggio partendo dai componenti più piccoli e via fino a i componenti più grandi. Le foto finali dei moduli sono mostrate nelle figure 7 e 8. Per i collegamenti tra i vari moduli seguite il piano di cablaggio in figura 9.

Figura 6 *Piano di montaggio del modulo RELBIN*

Per l'alloggiamento dei moduli nel contenitore potete seguire la disposizione mostrata nella foto in figura 10.

Nelle figure 11 e 12, invece, trovate le foto dei pannelli del mio prototipo.

Nel caso in cui decidiate di adottare un contenitore diverso, vi consiglio di:

1. Scegliere un contenitore dotato di feritoie o di griglie di aerazione.
2. Dotare il L298 di un buon dissipatore di calore. A causa della posizione del componente potreste avere qualche difficoltà nel trovare il dissipatore adeguato. In questo caso vi consiglio di realizzare un dissipatore a partire da una lastrina di alluminio di 2 o 3 mm di spessore, magari piegata ad L.
3. Disporre prese e spine nella parte posteriore del contenitore, in modo da evitare affollamenti del pannello anteriore e di migliorare la visibilità dei LED e del tasto di ESTOP (figure 11 e 12).
4. Per il connettore per i binari potete usate una di quelle morsettiere per casse acustiche. Sono molto comode ed hanno un prezzo irrisorio.
5. Inserite un fusibile di protezione sul segnale di alimentazione. Può sembrare una precauzione eccessiva, ma visti i prezzi delle locomotive e degli scambi, è meglio abbondare in protezioni.

Una volta chiuso il contenitore dovete ancora fare due scelte: l'alimentatore e il cavo di connessione con la linea seriale del PC.

Figura 8 *Il modulo RELBIN*

Nel mio caso la scelta è caduta sull'alimentatore di un notebook.

Vi ricordo che la tensione di alimentazione non dovrebbe superare i 18volt, inoltre, per pilotare certi scambi occorrono delle correnti di almeno 1ampere, quindi l'alimentatore dovrebbe essere in grado di fornire almeno 3 ampere, per evitare che quando gli scambi vengono azionati vi sia un calo di tensione sulle rotaie, con conseguenti rallentamenti delle locomotive.

Figura 7 *Il modulo HBOOST*

Per il cavo RS232 basta usare uno di quelli 1 a 1 con un connettore femmina (lato PC) e uno maschio (lato TDS). Potete anche realizzarlo da soli seguendo lo schema in figura 13.

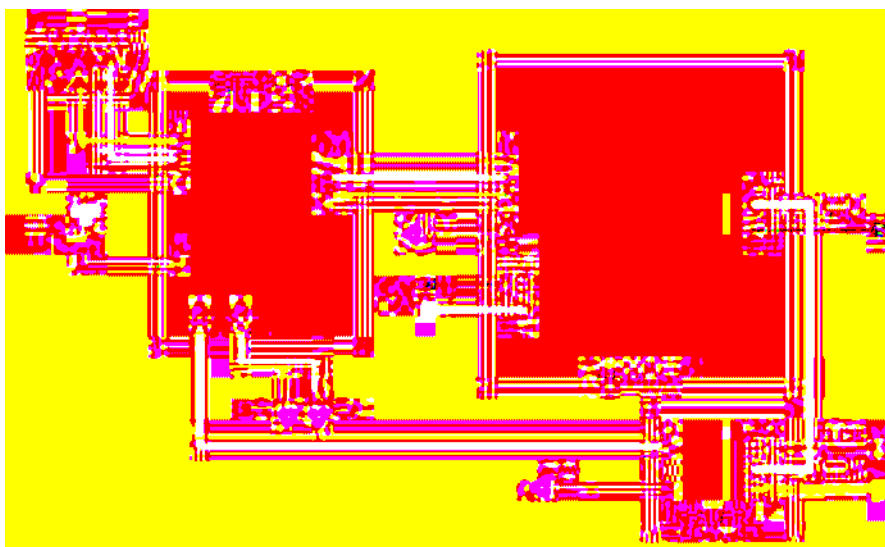



Figura 9 Collegamenti tra i moduli

IL SOFTWARE DI CONTROLLO

Il software di controllo (scaricabile dal sito di fare elettronica), è composto di quattro file: TDS.EXE, TDS.ICO, TDP.HLP e TDS.CFG. Il primo è l'eseguibile, compatibile con tutti i sistemi operativi a partire da Windows 95, il secondo è un file di configurazione contenete i parametri per il controllo del sistema.

Dopo aver scaricato e copiato i file in una directory del PC, lanciate il file TDS.EXE. Alla partenza il software tenterà di configurare la porta seriale e in caso di successo invierà il primo messaggio a TDS per verificare la sua presenza. Qualunque errore durante questa sequenza verrà segnalato dalla comparsa di una finestra.

Nell'ipotesi che la linea seriale sia stata configurata e TDS sia connesso e acceso, comparirà una finestra (figura 14) con una barra dei menu e due pulsanti per lo start e lo stop del programma. Di seguito vi descriverò una sessione di esecuzione a partire dalla configurazione dei parametri del sistema. In tabella 1 trovate un elenco dei menu con la descrizione delle loro voci.

Nel caso in cui si sia verificato un errore alla partenza, controllate che TDS sia accesso e la linea seriale sia connessa. Il programma alla partenza tenta di configurare la porta "Com1" se non ci riesce vi invia un messaggio di errore. In questo caso dovete configurare la porta

Figura 10 I moduli alloggiati nel contenitore

Figura 11 Pannello Frontale del sistema

Figura 12 Pannello posteriore del contenitore

Figura 13 *Realizzazione del cavo RS232*

manualmente, selezionando la voce "comunicazione" nel menu delle proprietà. Comparirà una finestra (figura 15) con un box di selezione della porta (Com1 o Com2) e un box per la scelta del periodo di invio dei comandi. Questo valore influisce sulla velocità di invio dei messaggi sulla linea seriale. Il valore 100 è stato scelto dopo una serie di prove con vari PC e vari sistemi operativi.
Se durante l'utilizzo del sistema, notate delle anomalie di comunicazione (segnalate da appositi messaggi a video) provate ad aumentarlo (ad esempio a passi di 50) fino a trovare il valore ottimale.

La configurazione

Il primo passo è quello di configurare le locomotive. Nel menu proprietà selezionate la voce "Locomotive". Comparirà una finestra con diversi controlli (figura 17). In questa finestra potete selezionare una delle possibili locomotive, assegnargli un indirizzo, definire un nome, scegliere il numero delle funzioni e i passi di velocità.

Figura 14 *Finestra principale del software*

Figura 15 *Finestra delle proprietà di comunicazione*

Una volta impostato il tutto premete il pulsante "salva".
L'indirizzo ha una caratteristica particolare: se posto a zero disabilita l'invio dei comandi. Questa caratteristica è utile se avete meno di quattro locomotive, in questo caso basta porre a zero l'indirizzo delle locomotive eccedenti, per escluderle dal sistema. Ripetete i passi precedenti per ciascuna locomotiva.

Figura 16 *Finestra delle proprietà DCC*

A questo punto passiamo agli accessori fissi. Selezionando la voce "Accessori" nel menu proprietà comparirà una finestra di dialogo con tutte le opzioni per la configurazione degli accessori (figura 17).
Anche qui abbiamo la possibilità di assegnare: il nome, l'indirizzo, la coppia (vi ricordo che gli accessori sono accoppiati a due a due per consentire il controllo degli scambi), il nome dello stato "on" e il nome dello stato "off". Per l'indirizzo vale quanto detto per le locomotive, se posto a zero disabilita l'accessorio. Anche qui ripetete i passi precedenti per ciascun accessorio.
Completata questa sequenza potete salvare la configurazione, selezionando la voce "Salva configurazione" nel menu file. A questo punto il sistema è pronto all'uso.

Figura 17 *Finestra delle proprietà delle locomotive*

Figura 18 *Finestra delle proprietà degli accessori*

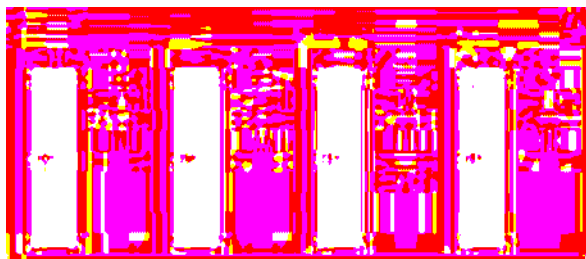



Figura 19 *Pannello di comando delle locomotive*

La fase operativa

Per entrare in questa fase basta premere il pulsante "START".
Dopo l'attivazione compariranno due finestre: una per il controllo delle locomotive (figura 19) e una per gli accessori (figura 20). Tutti i dispositivi non presenti (cioè quelli con l'indirizzo a zero) saranno disattivati. Sui controlli presenti in queste finestre non credo che sia molto da dire.
L'unica particolarità è il contatore nella finestra principale che avanzerà in seguito dell'invio di un comando.
Questa informazione può essere utile per capire se il software sta inviando messaggi sulla linea seriale. Durante la fase operativa tutte le voci di menù vengono disabilitate ad eccezione di quella per l'uscita dal programma e di quelle di help.

Figura 20 *Pannello di comando degli accessori*

La fase di servizio

Le procedure per configurare le CV sono associate alle voci del menù "servizio".
Per scrivere o verificare una CV, si deve selezionare la voce "Programma CV".
Verrà visualizzata una finestra di dialogo (figura 21) dove dovremo inserire: l'indirizzo della variabile, il valore desiderato e il tipo di operazione (scrittura o verifica).
A questo punto premendo il pulsante "Programma" verrà avviata la sequenza di programmazione.
Vi ricordo che in questo caso il segnale DCC viene inviato sul binario di programmazione, quindi prima di iniziare assicuratevi che il decoder da programmare sia in "contatto" con esso.

Figura 21 *Finestra per la programmazione di una CV*

Il termine dell'operazione di programmazione sarà segnalato dalla comparsa di un finestra di dialogo con un messaggio sul risultato dell'operazione.
Questa segnalazione si basa sull'utilizzo del meccanismo di Acknowledge, ma alcune volte può succedere che il decoder non sia in grado di emetterlo, oppure lo emetta in maniera errata (alcuni decoder per accessori non rispettano le specifiche DCC riguardo questo aspetto). In questo caso prima di eseguire la programmazione, selezionate la voce "Dcc" del menu proprietà. Vi verrà mostrata una finestra di dialogo (figura 16) che vi permetterà di disabilitare il meccanismo di ricezione dell'Acknowledge (ponendo il controllo su "Manuale").
Le altre due voci del menù ("Ricerca Indirizzo" e "Ricerca Valore") verranno utilizzate per la ricerca dell'indirizzo di una locomotiva oppure per la ricerca del valore contenuto in una CV. Ovviamente le procedure di "auto ricerca" verranno disabilitate nel caso in cui sia stata selezionata la modalità manuale per la ricezione dell'Acknowledge.

Gli errori

Durante l'utilizzo del sistema si possono verificare degli errori, dovuti ad una errata comunicazione oppure a "incidenti" sul tracciato (ad esempio un convoglio che deraglia può mettere in cortocircuito le rotaie, causando un assorbimento eccessivo da parte del booster).
In caso di errori il software sospende l'invio dei comandi seriali e mostra una finestra infor-

MENU	VOCE	DESCRIZIONE
File	Salva Configurazione	Salva la configurazione corrente nel file TDS.cfg. Dopo aver modificato la configurazione del sistema il salvataggio dei nuovi valori non viene effettuato in maniera automatica, quindi ricordatevi di effettuare questa operazione.
	Exit	Esce dal programma.
Proprietà	Comunicazioni	Permette di configurare la porta di comunicazione (Com1, Com2) e il periodo di invio dei comandi (Tra i 10 e i 30msec).
	Dcc	Permette di configurare il parametro AUTO_ACK del protocollo DCC: spuntando la voce AUTO le operazioni di programmazione attendono il segnale di conferma; spuntando la voce MANUALE, l'operazione viene eseguita senza nessun controllo. Il secondo caso può essere utile quando abbiamo dei decoder che non gestiscono la risposta.
	Locomotive	A questa voce è associata un finestra di dialogo che consente di configurare i parametri di ciascuna locomotiva. I parametri sono i seguenti: ■ Nome della locomotiva. ■ Passi di velocità: 14 o 28. ■ Numero delle funzioni: da 1 a 4.
	Accessori	In questa finestra di dialogo potete configurare ciascun accessorio.
Servizio	Programma CV	A questa voce è associata una finestra di dialogo che consente di eseguire le operazioni di programmazione su una CV definita dall'utente. La finestra presenta: ■ Un casella di input dove inserire l'indirizzo della CV. ■ Un casella di input dove inserire il valore della CV. ■ Un casella di input di scelta per il tipo di operazione: Scrittura o verifica. ■ Un pulsante che consente di avviare l'operazione. ■ Un pulsante per l'uscita.
	Ricerca Indirizzo Locomotiva	Questa voce consente di ricercare l'indirizzo di una locomotiva. Il funzionamento si basa sul riconoscimento del segnale di acknowledge da parte del booster. La voce viene disabilitata se il parametro ACK e posto in modalità manuale nella finestra delle proprietà DCC.
	Ricerca Valore	Stesso discorso del caso precedente. In questo caso viene ricercato il valore contenuto in una CV, il cui indirizzo è stato specificato dall'utente.
Help	Help About Hw	Visualizza l'help del programma. Visualizza le versioni dell'hardware e del firmware dell'unità.
	About TDS	Fornisce informazioni circa il programma.

Tabella 1 *Menù e descrizioni delle voci associate*

mativa con il tipo di errore rilevato.
I messaggi di errore, i loro significati e le azioni da eseguire per rimuoverli sono elencate nella tabella 2.

CONCLUSIONI

Il sistema funziona da diversi mesi, su un plastico dove abitualmente girano 3 locomotive, 6 scambi, 3 semafori e una serie di luci.
Il software può essere migliorato soprattutto nella parte di gestione degli accessori e nella gestione della modalità di servizio.
In questo caso si potrebbe includere un database con i formati di tutte le CV in modo da poter guidare l'utente nella configurazione dei decoder.
Tutte queste varianti saranno incluse in una prossima versione.
buon divertimento!

Figura 22 *Finestra per la ricerca dell'indirizzo di una locomotiva*

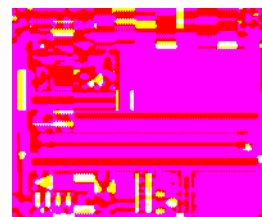



Figura 23 *Finestra per la ricerca del valore di una CV*

Messaggio	Significato	Azione
"Comunicazione Ok"	TDS è acceso e funzionante.	Nessuna.
"Porta Seriale non aperta" "Porta Seriale non chiusa"	La porta seriale scelta per le comunicazioni non è disponibile.	Selezionare la voce di menù "Proprietà COM" e cambiare la porta seriale in uso.
"Nessuna Risposta"	TDS non ha risposto ad un messaggio di comando.	TDS potrebbe essere spento oppure non collegato alla linea seriale. Verificate le alimentazioni e le connessioni 232.
"Errore di trasmissione"	Il TX non ha trasmesso tutti i byte del messaggio.	Spegnere tutto e provare a riavviare il sistema. Questo problema si verifica in genere quando il PC trasmette troppi messaggi. In questo caso provate ad aumentare il valore contenuto nel box "periodo" della finestra di dialogo associata alla voce di menù "Proprietà COM".
"Booster in sovraccarico"	Il BOOSTER è andato in corto oppure ha fornito troppa corrente.	Spegnere tutto e controllare il plastico.
"Stop di emergenza"	L'utente ha premut il tasto ESTOP.	
"Messaggio sconosciuto" "Messaggio di risposta Errato"	Il messaggio di comando inviato è sconosciuto oppure il messaggio di risposta è errato.	Il sistema ha ricevuto o inviato un messaggio errato oppure c'è un problema nel cavo 232. In questo caso spegnete tutto e verificate il cavo 232.

Tabella 2 *Messaggi di errore*

L'amplificatore operazionale dalla A alla Z

Prosegue, in questa seconda parte, l'esposizione dei parametri che maggiormente caratterizzano gli amplificatori operazionali

Nel precedente articolo sono già stati esaminati alcuni parametri fondamentali dell'amplificatore operazionale: il guadagno di tensione ad anello aperto, la tensione offset di ingresso e le correnti di polarizzazione e di offset. Ma vi sono altri parametri elettrici che per quanto riguarda il comportamento dell'op-amp sia in corrente continua che in corrente l'alternata, sono da prendersi nella dovuta considerazione quando si vuole progettare un qualsiasi circuito che utilizzi questo componente.

2.1 IL RAPPORTO DI REIEZIONE AL MODO COMUNE (CMRR)

Se ai due ingressi di un op-amp si applica, in continua o in alternata, il medesimo segnale, definito quindi segnale di modo comune, la tensione di uscita, che dovrebbe essere nulla [infatti, essendo $V_o = A_{OL} (V_a - V_b)$, per $V_a = V_b$ dovrebbe essere $V_o = 0$], ha un valore minimo ma comunque diverso da zero. Si definisce guadagno di modo comune A_{cm} il rapporto fra la tensione V_o di uscita e la tensione V_{imc} applicata a entrambi gli ingressi *a* e *b* (tensione di ingresso di modo comune). Si ha quindi:

$$A_{cm} = V_o / V_{imc}$$

Essendo $V_o << V_{imc}$, il guadagno di modo comune sarà sempre un numero piccolo. Nei manuali viene riportato il cosiddetto rapporto di reiezione al modo comune (*CMRR* = *Common-Mode Rejection Ratio*) espresso quindi dal rapporto:

$$CMRR = A_{OL} / A_{cm} \quad [2.1]$$

Esprimendo quest'ultimo in dB si ha:

$$CMRR\ (dB) = 20 \log (A_{OL} / A_{cm}) = 20 \log CMRR \quad [2.2]$$

Poiché A_{OL}, guadagno di tensione ad anello aperto, ossia senza alcuna connessione di reazione fra uscita e ingresso, è molto grande rispetto ad A_{cm}, il *CMRR* è rappresentato sempre da un numero molto elevato. Valori di 120 dB sono ormai usuali. L'LM833 ha un *CMRR* tipico di 100 dB e minimo di 80 dB. Il *CMRR* è un parametro altamente qualitativo dell'amplificatore operazionale. Tanto più elevato è il *CMRR* tanto meno l'op-amp, nella sua funzione amplificatrice, risentirà della tensione di modo comune che per un qualsiasi motivo (rumore di natura elettrica, di natura elettromagnetica, ecc.), potrà presentarsi ai suoi ingressi. In termini semplici questo parametro rappresenta la capacità dell'op-amp di cancellare più o meno totalmente un segnale che si presenti identico ad entrambi gli ingressi.

Il *CMRR* varia con la frequenza nel senso che all'aumentare di questa si mantiene costante entro un certo intervallo per poi decrescere secondo una curva che ha pendenza di 20 dB per decade, ossia di 20 dB per ogni decuplicazione della frequenza. Per la misura del *CMRR* si consideri il circuito di cui alla figura 2.1 nel quale ai due ingressi dell'operazionale LM833 è applicato lo stesso segnale V_i – segnale quindi di modo comune. Il guadagno A_{cm} dell'op-amp si può esprimere tramite il rapporto $\Delta V_o / \Delta V_{imc}$ dove V_{imc} è il segnale di modo comune applicato in ingresso e V_o è la tensione di uscita. Tramite la transcaratteristica

di Nico Grilloni
n.grilloni@farelettronica.com

Grandezze in C.A. e C.C. caratterizzanti l'op-amp

fornita da SPICE si ha il diagramma di cui alla figura 2.2 dal quale è facile calcolare il rapporto $\Delta V_o / \Delta V_{imc}$. È infatti sufficiente portare i marker *a* e *b* su due punti qualunque del tratto in cui la transcaratteristica è lineare e leggere i corrispondenti valori della tensione di uscita indicati dalla posizione dei marker orizzontali *c* e *d*. In corrispondenza della tensione di ingresso indicata da $Xb = -2{,}926$ V, si legge una tensione di uscita $Yd = -12$ V, mentre in corrispondenza della tensione di ingresso indicata da $Xa = 4{,}815$ V si legge una tensione di uscita $Yc = +12{,}15$ V. Si ha quindi:

$$\Delta V_o = -12 - 12{,}15 = -24{,}15 \text{ V}$$

$$\Delta V_{imc} = Xb - Xa = -2{,}926 - 4{,}815 = -7{,}741 \text{ V}$$

Per A_{cm} si ha quindi:

$$A_{cm} = 24{,}15 \;/\; 7{,}741 = 3{,}119$$

Tenendo conto che l'op-amp utilizzato in simulazione è l'LM833 per il quale il costruttore specifica un guadagno ad anello aperto $A_{OL} = 110$ dB a cui corrisponde un valore numerico pari a $10^{5,5}$, dall'espressione [2.2] si ricava:

$$CMRR = 20 \lg (10^{5,5} \;/\; 3{,}119) = 100{,}1 \text{ dB}$$

Per l'LM833 il costruttore specifica, infatti, come si è detto, un valore del *CMRR* pari a 100 dB.
Il *CMRR* si può ricavare anche utilizzando un segnale di modo comune alternato attraverso la curva di risposta in frequenza. Sostituendo quindi al generatore della figura 2.1 un generatore di tensione sinusoidale alla frequenza di 100 Hz, si ottiene in simulazione il diagramma di risposta in frequenza di cui alla figura 2.3. In questo si vede che il guadagno A_{cm} di modo

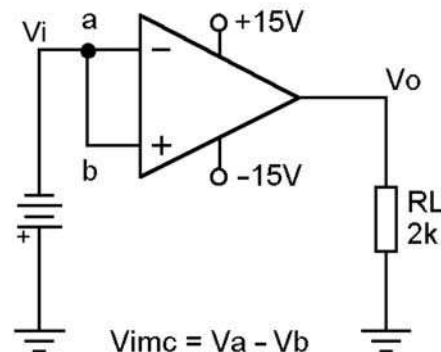



Figura 2.1 *Configurazione adatta alla misura del CMRR dell'op-amp*

comune, coincidente con l'ordinata *Yc*, vale 9,819 dB a cui corrisponde il valore numerico 3,1 già in precedenza ricavato tramite la transcaratteristica dell'op-amp. Applicando quindi l'espressione [2.2] si perviene al valore del *CMRR*. Nella determinazione del *CMRR* di un operazionale è opportuno tenere presente le condizioni e la configurazione spesso suggerite dalle Case e rammentare, nel caso si faccia ricorso a un segnale di tipo sinusoidale, di impostare il relativo generatore su una frequenza non superiore a 100 Hz. Nella maggior parte dei casi, infatti, oltre questa frequenza il *CMRR* diminuisce secondo una curva di pendenza pari a 20 dB/decade. Per operazionali veloci è comunque possibile utilizzare frequenze anche più elevate.

2.2 IL RAPPORTO DI REIEZIONE ALLA TENSIONE DI ALIMENTAZIONE (PSRR)

La maggior parte dei circuiti integrati lineari è sensibile alle variazioni della tensione di alimentazione. Queste variazioni sono da evitarsi o, quantomeno da contenersi, dal momento che determinano alterazioni della tensione di offset di ingresso. Indicando con V_{al} la generica tensione di alimentazione (V_{CC} e V_{EE}), il rapporto:

$$PSRR = \Delta V_{os} / \Delta V_{al} \quad (\mu V/V) \qquad [2.3]$$

fra la variazione della tensione di offset e la variazione della tensione di alimentazione V_{al} che l'ha prodotta, si definisce rapporto di reiezione alla tensione di alimentazione (*Power Supply Rejection Ratio*). Nella manualistica è prevalentemente espresso in dB e, talvolta, in μV/V. L'espressione del *PSRR* in dB è:

$$PSRR\,(dB) = 20 \log (\Delta V_{al} / \Delta V_{os}) = 20 \log = (1/PSRR) \qquad [2.4]$$

Per esempio, per l'LM833 che ha un *PSRR* = 100 dB, dalla [2.3] si ha:

$$100 = 20 \log (\Delta V_{al} / \Delta V_{os})$$

Da questa si ricava:

$$10^5 = (\Delta V_{al} / \Delta V_{os}) = 1/PSRR$$

Pertanto per il *PSRR* espresso in μV/V si ha:

$$PSRR = 1/10^5 = 10 \cdot 10^{-6} = 10\ \mu V/V$$

Ciò significa che per ogni volt di variazione della tensione di alimentazione si hanno 10 μV di variazione della tensione di offset di ingresso dell'op-amp. I migliori op-amp saranno quelli a più basso *PSRR* se espresso in μV/V o a più alto *PSRR* se espresso in dB. Valori di 100 dB e anche più elevati sono ormai frequenti.

2.3 ASSORBIMENTO DI CORRENTE

È la corrente assorbita dall'amplificatore operazionale per il suo normale funzionamento. È misurata in assenza di carico e con tensione di uscita nulla.

2.4 PARAMETRI INERENTI LE CARATTERISTICHE IN CORRENTE ALTERNATA

2.4.1 La velocità di risposta (Slew Rate)

Lo *slew rate* (*SR*) indica la velocità di risposta dell'amplificatore operazionale, ossia la sua capacità di fornire in uscita un segnale pressoché eguale al segnale di ingresso. Se il segnale V_i ha un'ampiezza molto elevata o presenta un fronte di salita brusco (come nel caso, per esempio, di un'onda quadra), in uscita si può avere un segnale fortemente distorto così come mostra la figura 2.4 nella quale si può notare il ritardo del segnale di uscita (OUTPUT) rispetto al segnale a onda quadra applicato in ingresso (INPUT). Lo *slew* rate ha l'espressione:

$$SR = \Delta V_o / \Delta t \ (V/\mu s) \qquad [2.5]$$

Il valore dello *slew rate* viene sempre fornito nei fogli tecnici per un guadagno unitario: l'operazionale LM833, per esempio, ha uno *slew rate* di 7 V/μs per un guadagno unitario, ma vi sono operazionali, definiti veloci, con velocità di risposta anche superiore a 2000 volt per microsecondo (e anche dell'ordine di 6000 V/μs).

In particolare, la conoscenza dello *slew rate* è essenziale quando si debbano amplificare segnali di ampiezza notevole e/o ad andamento impulsivo.

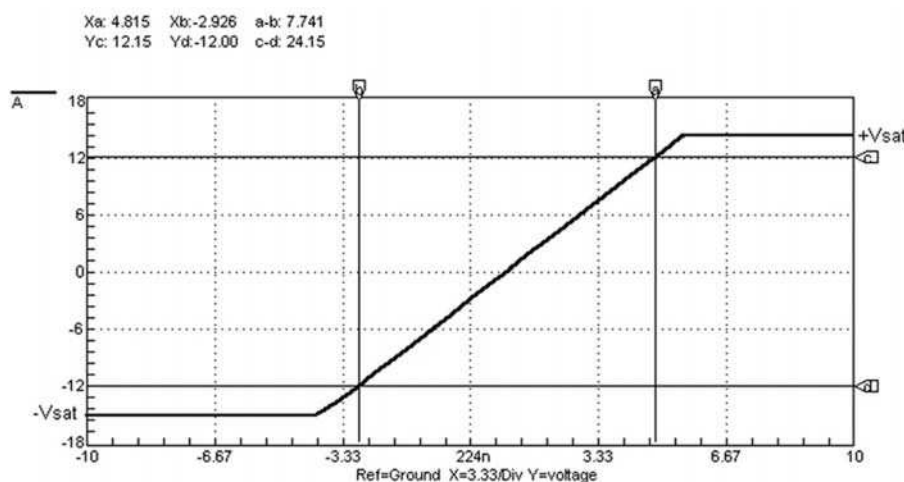



Figura 2.2 *Determinazione del CMRR attraverso la caratteristica di trasferimento*

Con riferimento a un segnale sinusoidale, nota la rapidità di risposta dell'op-amp e il valore di picco V_{oM} del medesimo segnale (eguale, essendo il guadagno unitario, al valore di picco del segnale in ingresso), si può calcolare la massima frequenza f_p (*full power response frequency*) compatibile con un determinato operazionale oltre la quale si ha in uscita un segnale fortemente distorto. L'espressione utile per il calcolo della frequenza f_p si ricava considerando in ingresso un segnale sinusoidale $V_i = V_{iM}$ sen ωt.
Per un guadagno pari a 1 si avrà $V_o = V_i$ e quindi:

$$V_o = V_{oM} \text{ sen } \omega t$$

Con alcuni calcoli, che qui si tralasciano, si ricava per lo slew rate l'espressione:

$$SR = \omega \cdot V_{oM} \cos \omega t$$

Il valore massimo di SR ha espressione:

$$SR = \omega \cdot V_{oM} = 2 \cdot \pi \cdot f_p \cdot V_{oM} \quad \text{(V/s)}$$

e quindi

$$SR = 2 \cdot \pi \cdot f_p \cdot V_{oM} \cdot 10^{-6} \quad \text{(V/µs)}$$

da cui

$$f_p \text{ (Hz)} = SR / (2 \cdot \pi \cdot V_{oM} \cdot 10^{-6}) = (SR \cdot 159000) = = / V_{oM} \qquad [2.6]$$

Supponendo, per esempio, un valore di picco V_{oM} = 0,5 V del segnale sinusoidale, utilizzando l'LM833 (SR = 7 V/µs) a guadagno unitario, si avrà un'amplificazione priva di distorsione fino alla frequenza:

$$f_p = (7 \times 159000) / 0{,}5 = 2{,}22 \text{ MHz}$$

Ciò significa che applicando un segnale di 0,5 V in ingresso con frequenza maggiore di 2,22 MHz si avrà in uscita un segnale distorto. In particolare, un segnale sinusoidale in ingresso, in uscita tenderà ad una forma d'onda triangolare.
Dalla [2.6] si ricavano le formule inverse:

$$SR \text{ (V/µs)} = (f_p \cdot V_{oM}) / 159000 \qquad [2.7]$$

$$V_{oM} \text{ (V)} = (SR \cdot 159000) / f_p \qquad [2.8]$$

Lo *slew rate*, vale ripeterlo, rappresenta un parametro da tenersi in particolare considerazione quando l'op-amp debba essere pilotato da segnali di notevole ampiezza e frequenza. In queste condizioni, non rispettando il valore dello *slew rate*, si ha in uscita un segnale più o meno distorto.
Nel caso di un segnale sinusoidale si è visto, per esempio, come la frequenza massima del segnale di ingresso f_p sia correlata all'ampiezza del segnale di uscita dall'espressione:

$$f_p = (SR \cdot 159000) / V_{oM}$$

Nel caso dell'operazionale LM833 il cui *slew rate* è di 7 V/µs per un guadagno unitario, si è calcolata una frequenza massima f_p = 2,22 MHz per un valore di picco del segnale di uscita V_{oM} = 0,5 V.
La verifica di questo risultato attraverso la simulazione con SPICE, sarà proposta in seguito.
Qui consideriamo il circuito di cui alla figura 2.5a nel quale l'amplificatore operazionale è pilotato da un segnale a onda quadra di

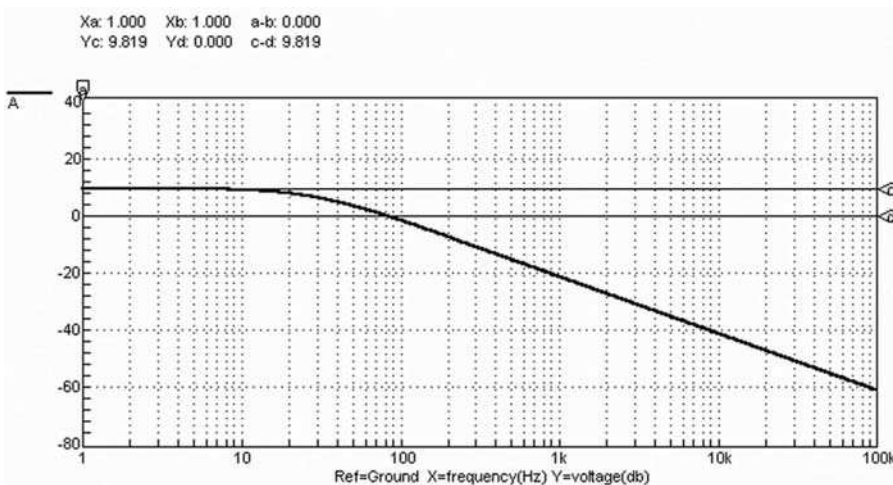



Figura 2.3 *Curva di risposta del circuito di cui alla figura 2.1 dopo aver sostituito il generatore in continua con un generatore in alternata. L'ordinata Yc = 9,819 dB indica il guadagno Acm.*

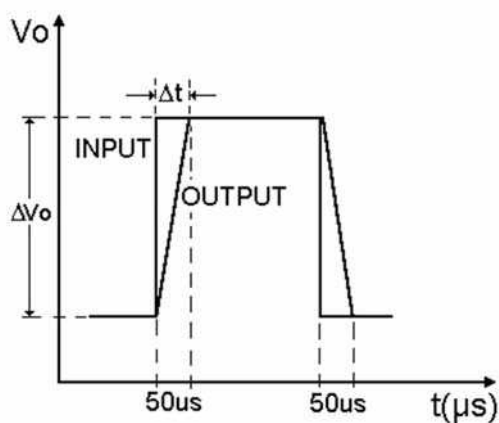



Figura 2.4 Se l'op-amp ha un basso slew rate e il segnale di ingresso è di ampiezza elevata, il segnale di uscita si presenterà distorto.

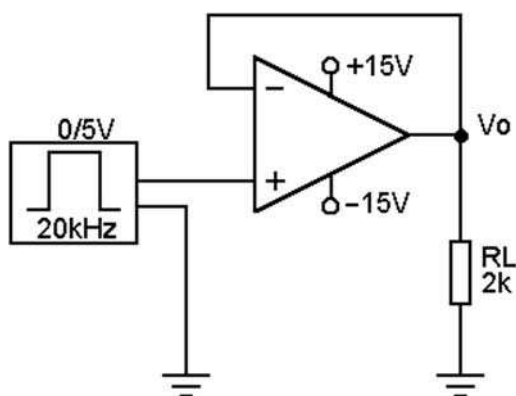



Figura 2.5a Amplificatore a guadagno unitario pilotato da un segnale a onda quadra per la verifica dello slew rate dell'op-amp utilizzato

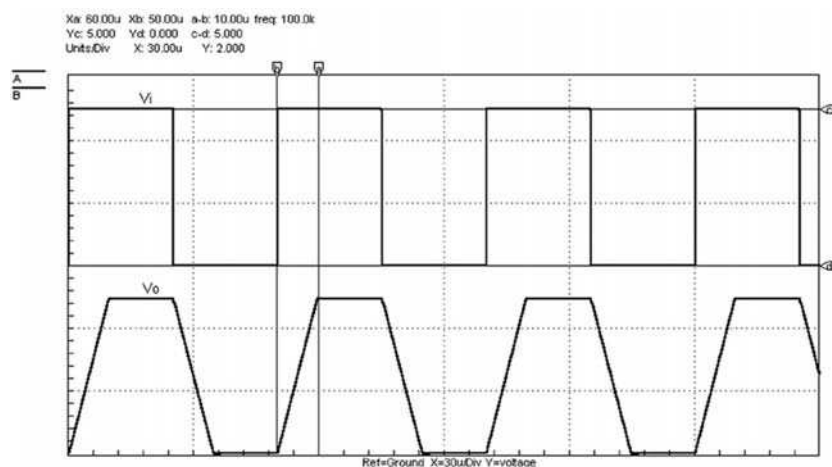



Figura 2.5b Andamento delle tensioni di ingresso e di uscita nell'operazionale LM747 utilizzato nel circuito della figura 2.5a per la verifica dello slew rate.

frequenza f = 20 kHz e di ampiezza pari a 5 V. La figura 2.5b riporta quindi il risultato della simulazione con SPICE. È evidente come sia sul fronte di salita che sul fronte di discesa la tensione di uscita presenti un certo ritardo rispetto alla tensione di ingresso. In pratica il segnale di uscita è fortemente distorto dal momento che i due fronti, perfettamente verticali nel segnale di ingresso, sono in uscita notevolmente inclinati.
La distanza (*a* – *b*) fra i marker verticali indica il ritardo, rispetto all'ingresso, con cui l'uscita raggiunge il valore massimo. Si legge (*a* – *b*) = 10 µs. Ciò significa, in base alla [2.5], che, essendo V_o = V_i = 5 V, lo *slew rate* vale 0,5 V/µs. Questo valore è proprio il valore tipico fornito dal costruttore per l'operazionale LM747 utilizzato in simulazione nel circuito di cui alla figura 2.5a.

2.4.2 Prodotto guadagno per banda passante (Gain Bandwidth Product)

È uno dei parametri più significativi dell'amplificatore operazionale dal momento che fornisce l'ampiezza della banda passante in funzione del guadagno di tensione A_{OL} secondo quanto illustrato nella figura 2.6 i cui diagrammi si definiscono curve di risposta. Si può constatare come la banda passante si contragga nel passaggio da un guadagno pari a 1 a un guadagno pari a 10 e quindi a 100. La frequenza di taglio f_H, che è quella frequenza alla quale il guadagno è pari a 0,707 volte il suo valore massimo (–3 dB), si porta infatti, all'aumentare del guadagno, da f_{H1} a f_{H10} e quindi a f_{H100} indicando una sempre più stretta banda passante.
Il prodotto del guadagno per la banda passante – indicato con *GBW* o *GBP* (*Gain Bandwidth Product*) o *UGB* (*Unity Gain Bandwidth*) – è costante ed è sempre rilevato con un carico R_L in uscita il cui valore è indicato dal costruttore. In pratica il *GBW*, sempre fornito nella manualistica, esprime l'am-

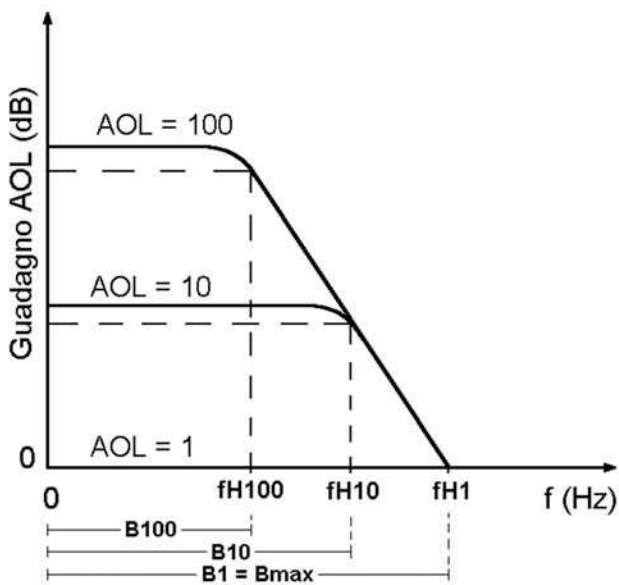



Figura 2.6 *All'aumentare del guadagno la banda passante dell'op-amp si contrae. L'ampiezza massima della banda (0 ÷ f_H) si ha per A_{OL} = 1 (0 dB)*

piezza della banda passante quando il guadagno è pari a 1 (da qui la definizione di *Unity Gain Bandwidth*), ossia la massima banda passante B_{max} prevista per un determinato operazionale che lavori con guadagno unitario. Per l'amplificatore operazionale LM833 la National Semiconductor, per esempio, fornisce un *GBW* minimo di 10 MHz e tipico di 15 MHz rilevato con ±15 V di alimentazione e un carico R_L = 2 kΩ. Ciò significa che questi valori di banda passante (10 e 15 MHz) si possono ottenere soltanto per un guadagno eguale a 1, mentre per guadagni crescenti si ha una contrazione via via crescente della banda passante. In simulazione al computer, con l'LM833 nella configurazione della figura 2.7, si è ottenuto il diagramma di cui alla figura 2.8. Qui si vede che la banda passante è massima (10 MHz) per un guadagno unitario, ossia per $V_o = V_i$ [per V_o/V_i pari a 1, essendo log 1 = 0, si ricava A_{OL} (dB) = 0 dB (cfr. articolo precedente)], mentre all'aumentare del guadagno si contrae fino alla frequenza f_t = 28,6 Hz. Questa, indicata dal marker verticale *a* che interseca la curva a –3 dB dal marker orizzontale *d*, è la frequenza di taglio caratteristica dell'op-amp LM833 ad anello aperto. L'ampiezza della banda passante, ossia la frequenza di taglio superiore f_H corrispondente a un definito guadagno, si può ottenere con l'espressione:

$$f_H = GBW/A_{OL} \qquad [2.9]$$

Per esempio, poiché per l'LM833 è *GBW* = 10 MHz, volendo sapere fino a quale frequenza di taglio superiore si estende la banda passante per un guadagno pari a 20 dB (ossia per un rapporto V_o / V_i = 10), si avrà:

$$f_H = (10 \times 10^6)/10 = 1 \text{ MHz}$$

Nella figura 2.9, eguale alla figura 2.8, si è posto in evidenza quanto appena calcolato. Il marker orizzontale *c* è a distanza di 20 dB dal marker orizzontale *d* che si trova su 0 dB, ossia sul livello del segnale di ingresso. La distanza (*c* – *d*) fra gli omonimi marker indica pertanto il guadagno pari a 20 dB. Il marker verticale *a*, posto sull'intersezione della curva di risposta col marker *c*, indica l'estremo superiore della banda passante, ossia la frequenza di taglio che esibisce l'op-amp LM833 per un guadagno A_{OL} = 10. Questa frequenza è assai prossima a quella su calcolata. Si legge, infatti, in alto nel diagramma, X_a = 1,136 MHz. Ovviamente l'espressione [2.9] si presta a risolvere il problema inverso.
Desiderando, per esempio, una banda passante che si estenda fino al limite superiore della banda audio, ossia fino a f_H = 20 kHz, e volendo quindi sapere qual è il guadagno corrispondente

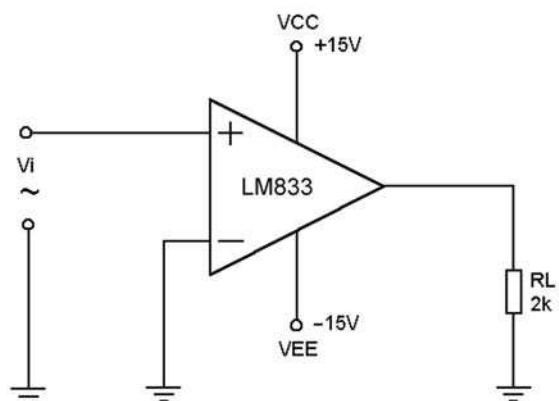



Figura 2.7 *Amplificatore con op-amp LM833 in configurazione non invertente tramite il quale si ricava la curva di risposta in frequenza una volta specificata la tensione di alimentazione e il carico RL.*

dello stadio amplificatore, si avrà:

$$A_{OL} = GBW/f_H \qquad [2.10]$$

ossia, sempre utilizzando l'LM833 per il quale A_{OL} = 10 MHz, si ricava:

$$A_{OL} = (10 \times 10^6) / 20000 = 500$$

Volendo esprimere questo guadagno in dB si ha:

$$A_{OL}\ (dB) = 20 \log 500 = 20 \times 2,7 = 54\ dB$$

L'amplificatore operazionale che qui si è utilizzato in simulazione al computer con il software SPICE, ha già un buon *GBW* particolarmente per le destinazioni d'uso indicate dalla Casa e, ancor più, se si pensa che l'ormai storico, ma sempre utilizzato LM741 ha un *GBW* di 1 MHz. Gli studi più avanzati nel settore hanno però condotto alla produzione di operazionali con valori del *GBW* che ormai superano il GHz estendendo quindi l'utilizzazione del componente all'alta frequenza.

2.5 ALTRI PARAMETRI

I dati tecnici relativi agli amplificatori operazionali forniscono, oltre alle grandezze su esposte, anche i valori di altri parametri strettamente inerenti la prevalente destinazione d'uso del componente. Per esempio, nel caso dell'LM833 che, come si è detto, in linea di massima è destinato all'utilizzazione in sistemi audio Hi-Fi, vengono riportati anche i valori del tasso di distorsione armonica totale espresso percentualmente, e la *Power Bandwidth* (quest'ultima indica la massima frequenza alla quale si può avere in uscita un valore di tensione di 10 V di picco senza distorsione).

2.5.1 La resistenza di ingresso (Input Resistance)

Non sempre ne sono forniti i valori. Nel caso ideale la resistenza di ingresso R_{in}, intesa come resistenza fra i pin invertente e non invertente, è infinita. In effetti si hanno valori, a temperatura ambiente, anche dell'ordine di 10^{12} Ω (valore caratteristico degli op-amp il cui stadio differenziale di ingresso è a Fet), ossia valori che, in pratica, si possono ritenere ideali. Il valore generalmente fornito dalle Case è il valore di resistenza misurato su un ingresso, mentre l'altro ingresso è a massa. Talvolta i due ingressi non presentano la stessa resistenza e allora la resistenza di ingresso può misurarsi come rapporto fra una variazione della tensione e la relativa variazione di corrente su un ingresso mentre l'altro ingresso è a massa.

2.5.2 La resistenza di uscita (Output Resistance)

La resistenza di uscita R_o, intesa come la resistenza vista dal terminale di uscita, è nulla nel caso ideale, mentre in realtà è dell'ordine di alcune decine o di qualche migliaio di ohm. Chiaramente quanto più sarà piccola la resistenza di uscita tanto più il comportamento dell'amplificatore operazionale sarà indipendente dal carico. A resistenze di uscita via via più elevate, e comunque confrontabili come ordine di grandezza con la resistenza di carico, corri-

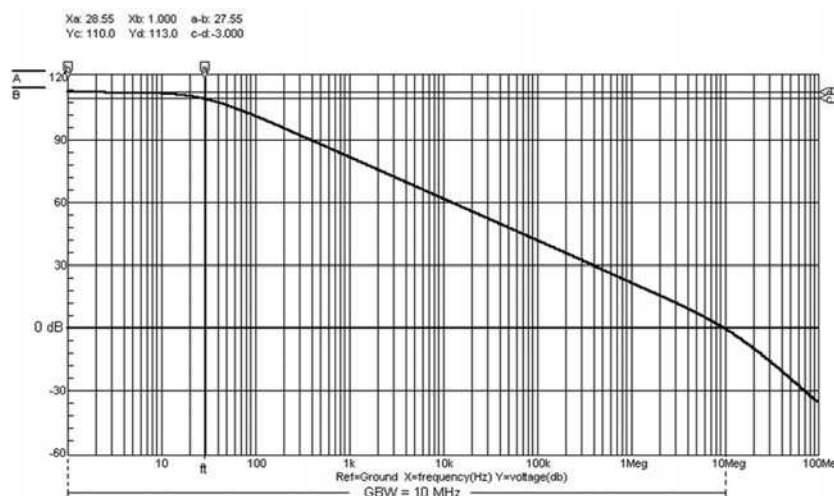



Figura 2.8 *Curva di risposta dell'amplificatore di cui alla figura 2.7. Il marker verticale a, posto all'intersezione della curva col marker orizzontale c, indica la frequenza di taglio f_H = 28,55 Hz (Xa = 28,55 Hz) caratteristica dell'LM833. Il marker orizzontale c è infatti a – 3 dB dal marker d. L'ordinata Yd esprime il guadagno pari a circa 113 dB possibile solo fra 0 Hz e 28,55 Hz.*

sponderà infatti una perdita di guadagno. Ciò non si verifica se il carico è almeno venti volte la resistenza di uscita dell'op-amp. Ad un amplificatore operazionale che abbia, per esempio, una resistenza di uscita di 100 Ω si applicherà quindi un carico di 2 kΩ o più elevato. Per misurare le resistenze di ingresso e di uscita è sufficiente ricorrere, rispettivamente, alle configurazioni illustrate nella figura 2.10a e nella figura 2.10b. La resistenza di ingresso R_{in} fornita dal costruttore è, infatti, il rapporto fra la tensione applicata a un ingresso dell'op-amp e la corrente che fluisce nel medesimo ingresso mantenendo l'altro ingresso a massa. Pertanto con riferimento alla figura 2.10a, avendo applicato la tensione di 1 V all'ingresso non invertente, posto in serie a questo il milliamperometro di SPICE e collegato a massa l'ingresso invertente, per la resistenza di ingresso si ha:

Figura 2.9 *La distanza (c – d) fra gli omonimi marker è pari a 20 dB. Il marker verticale a posto all'intersezione fra la curva di risposta e il marker c, indica la frequenza di taglio superiore per un guadagno di 20 dB. È infatti Xa = 1,136 MHz. Questo valore è molto prossimo a quello calcolato con l'espressione [2.9]*

$$R_{in} = 1 / (626{,}4 \times 10^{-9}) = 1{,}59\ \text{M}\Omega$$

essendo 626,4 nA la corrente indicata dal milliamperometro.

Applicata una tensione al pin di uscita, la resistenza di uscita sarà data dal rapporto fra questa tensione e la corrente corrispondente.

Per R_{out}, con riferimento alla figura 2.10b, si ha quindi:

$$R_{out} = 1/(21{,}17 \times 10^{-3}) = 47{,}2\Omega$$

essendo 21,17 mA la corrente indicata dal milliamperometro.

NOTE PER IL LETTORE

I diagrammi delle figure 3a e 3b sono stati ricavati al computer col software di simulazione SPICE (*Simulation Program of Integrated Circuits Emphasis*) – CircuitMaker – della Microcode Engineering Inc.

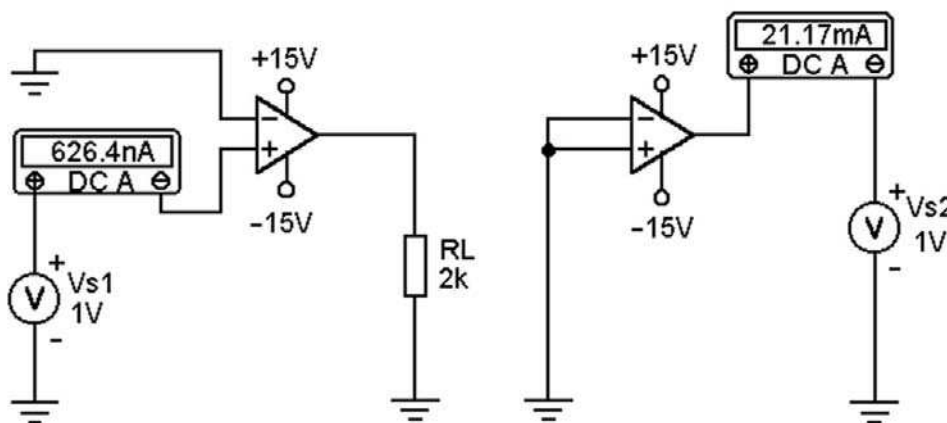



Figura 2.10 *In a il circuito per misurare la resistenza di ingresso e in b il circuito per misurare la resistenza di uscita di un op-amp*

Mikrobasic per PICmicro

Finalmente un corso molto semplice, ma allo stesso tempo efficace, per imparare a programmare i microcontrollori PIC, prodotti dalla famosa casa Microchip. Questa volta utilizziamo il linguaggio ad alto livello Basic, molto semplice ed intuitivo ma estremamente potente, che permette tempi di sviluppo e di manutenzione molto bassi, con il vantaggio di realizzazioni veloci e progetti immediati.

Dobbiamo proprio ammetterlo, l'elettronica è un settore ormai inarrestabile che, giorno dopo giorno, raggiunge livelli sempre più elevati nel campo della precisione, della velocità e della miniaturizzazione. Se cinquant'anni fa, in un metro cubo riuscivano a stare mille valvole, oggi, in un solo centimetro quadrato entrano comodamente miliardi di transistor molto più efficienti e veloci. Un esempio pratico? Pensiamo ai comodissimi *Pendrive* da 512 Mb, utili accessori che si collegano alle porte USB del computer e fungono da memorie di massa: ebbene, questi piccoli oggetti che stanno comodamente in tasca contengono circa 5 miliardi di "interruttori" capaci di memorizzare tantissime informazioni.
Chi si occupa d'elettronica deve oggi adeguarsi all'era **informatica** e **digitale**. È impossibile considerare le due cose separate. Basti pensare ai telefoni cellulari, ai moderni televisori, ai lettori DVD, alle centraline delle automobili, ai decoder satellitari e videogiochi: tutte queste apparecchiature non contengono al loro interno i normali circuiti elettronici, bensì sofisticati congegni debitamente **programmati** con opportuni linguaggi di programmazione.

L'ABITO NON FA IL MONACO

Andiamo a conoscere dunque questo strano componente, dall'aspetto umile ma misterioso, che non mancherà davvero di stupirci. Cos'è innanzitutto un microcontrollore? È un circuito integrato, di piccole dimensioni, la cui funzione non è stabilita a priori dal produttore, ma è programmata dall'utente finale o progettista. Un esempio ci chiarirà meglio il concetto: se acquistiamo un circuito integrato CD40106, possiamo solamente utilizzarlo quale sestupla porta NOT e nulla più, mentre un integrato CD4026 espleterà la sua funzione solamente come contatore e decoder LED. Essi non eseguiranno mai altre funzioni diverse da quelle per cui sono stati "programmati".
Invece un microcontrollore, se pur fisicamente identico ai suddetti IC, può compiere i più disparati compiti, sia semplici che enormemente complessi: tutto dipende dalla capacità del programmatore umano di impartire i comandi corretti, ossia di scrivere con un opportuno linguaggio, il **programma.** Facile a dirsi, difficile a farsi.
Nel corso di queste puntate daremo i primi passi verso questa difficile "arte", anticipando però già da ora che la *perfezione* si raggiungerà solo dopo un lungo e attento studio e sperimentazione.

PRIMO APPROCCIO

I PIC sono dei microcontrollori, che opportuna-

di Giovanni Di Maria
gdm1967@libero.it

La programmazione base

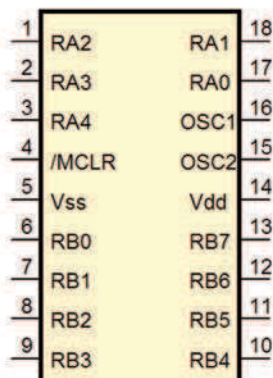



Figura 1 *Piedinatura del PIC16F84*

mente programmati, eseguono molto velocemente i compiti impartiti. In altri termini costituiscono un intero sistema programmabile digitale. Internamente sono costituiti da complessi dispositivi che somigliano molto, anche se in maniera ridotta, ad un personal computer.
Le parti fondamentali contenute sono: la **CPU** (Central Process Unit) che ha la funzione di eseguire una dopo l'altra le istruzioni programmate; la **PROM** (Progammable Read Only Memory) che contiene il programma scritto dall'uomo; la **EEPROM** (Electrical Erasable Programmable Read Only Memory) che può memorizzare i dati utente anche senza alimentazione elettrica; la memoria **RAM** (Random Access Memory) che contiene provvisoriamente i dati transitori e le variabili; le **PORTE** di Input e Output, con cui il PIC comunica con il mondo esterno e ad esse possono essere collegate qualunque tipo di apparecchiature, naturalmente con opportuni interfacciamenti elettronici.
Durante il nostro studio utilizzeremo il **PIC16F84** un microcontrollore molto popolare, costruito dalla **Microchip** (www.microchip.com), il quale presenta ottime caratteristiche di affidabilità, robustezza, anche alle inversioni di polarità, versatilità ed, oltretutto, è abbastanza economico. Inoltre può essere programmato praticamente un numero infinito di volte. Tale microcontrollore lavora con una tensione di 5 Volt CC. Non sono ammesse tolleranze e differenze elevate; in ogni caso è molto semplice ottenere questa tensione con opportuni regolatori.
Per un approfondimento sui microcontrollori PIC e la loro programmazione vi consiglio l'acquisto del libro *"Conoscere ed usare - PICmicro"* scritto da *Tiziano Galizia* e *Maurizio Del Corso* ed edito dalla *INWARE Edizioni.* Maggiori informazioni le trovate all'indirizzo www.farelettronica.com/libri.

LE FUNZIONI DEI 18 PIN

Il nostro PIC16F84 si presenta in contenitore DIL (dual in line) con 18 piedini, 9 per fila. Ogni pin ha un suo specifico compito, ma il fatto sorprendente è che, 13 di questi costituiscono le porte di ingresso e uscita: le funzioni "vitali" sono quindi svolte solamente dai rimanenti 5 piedini.
In figura 1 riportiamo la disposizione dei piedini (vista dall'alto dell'integrato).

Ogni piedino ha una sua funzione ben precisa:

- Il pin 14 (**VDD**) va collegato alla linea positiva dell'alimentazione (+5V).
- Il pin 5 (**VSS**) va collegata alla massa comune del circuito.
- Il pin 4 (**MCLR**) di norma va collegato al positivo dell'alimentazione; se riceve un breve impulso logico 0, viene eseguito il reset del chip, ed il programma inizia da capo (non preoccupatevi, non viene cancellato nessun dato).
- I pin 16 e 15 (**OSC1** e **OSC2**) servono per generare la cadenza del clock del micro. Approfondiremo bene questo concetto.
- I pin 6, 7, 8, 9, 10, 11, 12, 13 (**RB0-RB7**) costituiscono la PORTB, ossia 8 bit che si possono utilizzare per l'ingresso e l'uscita dei segnali logici.
- I pin 17, 18, 1, 2, 3 (**RA0-RA4**) costituiscono

la **PORTA**, ossia 5 bit che si possono utilizzare per l'ingresso e l'uscita dei segnali logici.

Tredici pin per la comunicazione con il mondo esterno sembrano moltissimi, ma vedrete che "l'appetito viene mangiando".

SCHEMA "BASE" APPLICATIVO

Il microcontrollore, per poter funzionare, deve essere collegato a pochissimi altri componenti esterni, che ne determinano le condizioni di funzionamento e soprattutto di velocità operativa. Sostanzialmente esistono due modi per fare ciò: una economica e "poco" precisa, l'altra meno economica ma molto precisa. Vediamoli entrambi.
Come detto prima, ai pin 15 e 16 vanno collegati alcuni componenti che determinano la velocità del clock e quindi di tutto il PIC. Scegliendo addirittura certi specifici valori, si può elevare il clock del micro (overclocking), ossia lo si può far funzionare a velocità maggiori di quelle previste dal chip stesso.
Chiaramente a nostro rischio e pericolo, in quanto l'intero circuito non garantirebbe quella precisione assicurata dalla casa produttrice e inoltre si avrebbe un consumo di corrente molto maggiore.

Collegamento con rete RC

L'assemblaggio più semplice è quello visualizzato in figura 2. Viene utilizzata una rete composta da una resistenza ed un condensatore per generare il segnale di clock. Si utilizza questa soluzione quando servono applicazioni dove la precisione non è fattore fondamentale.

Collegamento con rete QUARZATA

È sicuramente la configurazione preferita e precisa. Se si utilizza un quarzo da 4 Mhz, si otterrà dal software sviluppato, e quindi dal micro, una precisione dei tempi sbalorditiva, anche su quelli lunghi. In figura 3 è rappresentato lo schema base di funzionamento. Data la sua semplicità esso può essere assemblato tranquillamente su basetta mille fori. Si notino ai capi del quarzo i due condensatori ceramici da 22 pF.

Bene, abbiamo già abbastanza materiale per iniziare a lavorare. Entrambi gli schemi funzionano egregiamente, ma non sono capaci di visualizzare al mondo esterno i propri risultati e le elaborazioni, ne tanto meno, di compiere azioni meccaniche e simili. Più avanti vedremo come si fa. In ogni caso sappiate che questi schemi costituiscono già da soli una sorta di *Pic Test Board* con la quale si possono già effettuare i primi esperimenti e i test sul software sviluppato.

I LINGUAGGI DI PROGRAMMAZIONE

Per comunicare le proprie intenzioni ad un

Figura 2 *Schema BASE con rete RC*

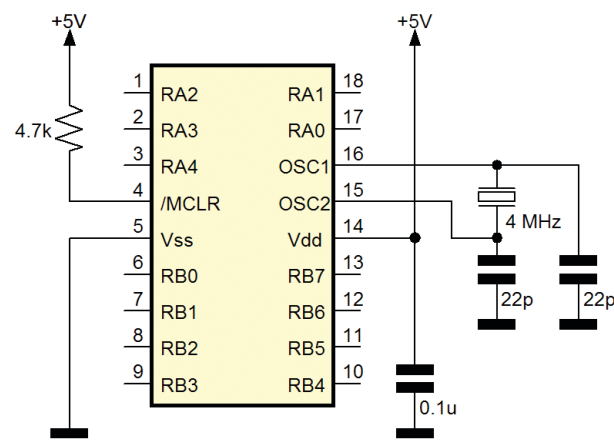



Figura 3 *Schema BASE con quarzo*

microcontrollore (o ad un computer) è necessario usare lo stesso linguaggio elettronico. Purtroppo esso è costituito solo da simboli elementari (0 e 1), davvero incomprensibili per gli esseri umani. Sono stati perciò ideati alcuni strumenti, in grado di tradurre i linguaggi con semantica umana in struttura binaria, che prendono il nome di interpreti e compilatori. Approfondiamo adesso il concetto di linguaggio di programmazione. Viene definito *programma* l'insieme sequenziale di comandi ed istruzioni lecite e corrette che la macchina esegue senza possibilità di errore al fine di risolvere un problema. A pensarci bene anche l'uomo, durante le sue azioni giornaliere, segue uno schema ordinato e sequenziale di compiti e doveri, predefiniti o meno che siano. Se vogliamo, in un esempio pratico, potremmo descrivere, con questo *pseudo-linguaggio*, le fasi salienti della sua giornata:

- Ore 07:00 ascolta la sveglia.
- Si alza dal letto.
- Va in bagno.
- Apre il rubinetto.
- **SE** l'acqua è fredda apre quella calda.
- Chiude il rubinetto.
- Va in camera da letto.
- Si veste.
- Va in cucina.
- Prepara il caffé.
- Versa il caffè nella tazza.
- Versa un cucchiaino di zucchero.
- **Ripete** l'azione dello zucchero per 3 volte.
- … e così via.

Come si può notare, anche nella nostra vita pratica, vengono seguite ed eseguite alcune azioni semplici, ma altre avvengono SOLO in certe situazioni (esempio dell'acqua calda), o altre ancora si ripetono per un certo numero di volte (esempio del cucchiaino di zucchero).
È proprio questo, il programma, con la "sola" differenza che per spiegare un comportamento ad un microcontrollore occorre la conoscenza di un linguaggio ad esso comprensibile.

Linguaggi di alto e basso livello

A seconda se i linguaggi utilizzati si avvicinano maggiormente al lessico umano o a quello elettronico, possiamo distinguerli in *alto* e *basso livello.* È linguaggio di alto livello quello che è simile alla lingua parlata umana (di solito inglese) con costrutti lessicali e semantici ad alto *livello*. È linguaggio a basso livello invece quello che somiglia alla codifica elettronica. In figura 4 vediamo questa suddivisione. Naturalmente entrambi presentano vantaggi e svantaggi. Vediamo quali. Il linguaggio ad alto livello è semplice da apprendere, è facilmente gestibile e la sua manutenzione non comporta problemi di sorta; è però lento da eseguirsi ed il codice generato occupa molta memoria elettronica. Ma il neo principale consiste nella perdita di tutte quelle finezze e quelle conoscenze di ciò che succede all'interno del controllore, in quanto sono già scontate e trasparenti. Il linguaggio a basso livello invece è molto difficile da imparare e l'apporto di modifiche al codice potrebbe risultare molto pesante; però la sua esecuzione è molto veloce ed il codice generato è piccolissimo. Per poter programmare con tale linguaggio è necessaria la **completa** conoscenza dell' hardware utilizzato. Se vogliamo fare un paragone, un po' spartano a dire la verità, è come se cucinassimo una busta di pasta già pronta (linguaggio ad alto livello) invece di prepararla completamente in casa (linguaggio a basso livello). La scelta dipende da come si vuole gestire il proprio tempo e denaro. I tempi di sviluppo del software, in un linguaggio ad alto livello, sono molto ridotti: in poco tempo un prototipo viene consegnato ed eventualmente modificato con pochissime conseguenze. Tra i linguaggi ad alto livello ricordiamo il Basic, il Pascal e il C mentre tra quelli a basso livello domina l'Assembler. In questo corso studieremo il semplice ma potente linguaggio **Basic.**

IL BASIC

È uno dei più "antichi" linguaggi di programmazione. È nato per dare ai principianti la possibilità di programmare gli elaboratori elettronici con facilità, senza il bisogno di conoscere a fondo la struttura dei processori. Infatti la parola Basic sta per **B**eginner's **A**ll Purposes **S**ymbolic **I**nstruction

Code (Codice simbolico di istruzioni per principianti adatto a tutte le applicazioni). È composto da un certo numero di comandi e istruzioni (in lingua inglese) che, nella loro sequenzialità, sono rigorosamente eseguiti dal microcontrollore. Tale linguaggio è una sorta di compendio, di sommario di tutto ciò che deve svolgere il processore. Nel corso delle puntate approfondiremo i vari concetti, considerando che, con l'elettronica di oggi, il tecnico deve assumere il duplice ruolo di **progettista** elettronico nonché di **programmatore** software. La capacità del programmatore consisterà pertanto nell'includere in maniera esatta tutte le funzioni che il processore dovrà svolgere, anche perché basterebbe un piccolissimo errore di codice a mandare in tilt l'intero sistema (bug).
Come vedrete in seguito, anche il più complesso programma, avrà sempre e solo uno scopo: quello di far cambiare, secondo le necessità del progetto, lo stato logico di una o più porte del PIC (0 Volt – 5 Volt).

IL MIKROBASIC

Per i nostri progetti e i nostri esperimenti utilizzeremo l'innovativo e potente mikrobasic, prodotto dalla **mikroElektronika**, azienda iugoslava. La nostra scelta è caduta su tale soluzione per alcuni motivi che di seguito elenchiamo:

- È un completo ed efficiente ambiente IDE (Integrated Development Environment), ossia in un unico pacchetto software sono compresi l'editor del sorgente, il compilatore Basic, il debugger ed il programmatore per scrivere sul nostro PIC).
- Dispone di un convertitore in notazione HEX, BIN e DEC (utilissimo).
- Dispone di un Code Explorer avanzato.
- Presenta on ottimo assistente di codice.
- Permette la colorazione del codice sorgente contestualmente alla sintassi del codice.
- Possiede un esauriente Help on-line.
- Genera automaticamente il file HEX.
- Visualizza il codice Assembler del programma.
- Mostra le statistiche grafiche dell'uso delle risorse.
- Presenta molte utilità.
- …. e molto ancora.

In figura 5 vediamo come si presenta la schermata principale del compilatore, la cui versione demo (limitata a 2K di codice) si può scaricare dal sito di Fare Elettronica (www.farelettronica.com).

ACCENDIAMO IL NOSTRO PRIMO LED

Vediamo adesso come, con semplici operazioni, è possibile far illuminare un diodo led collegato

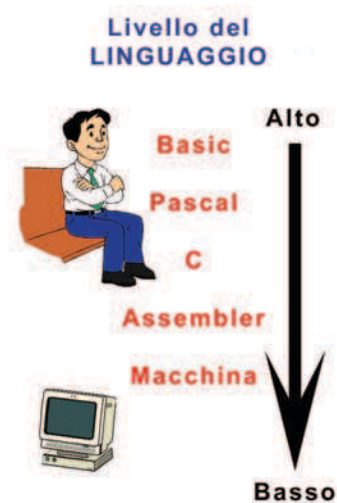



Figura 4 *Linguaggi di Alto e Basso livello*

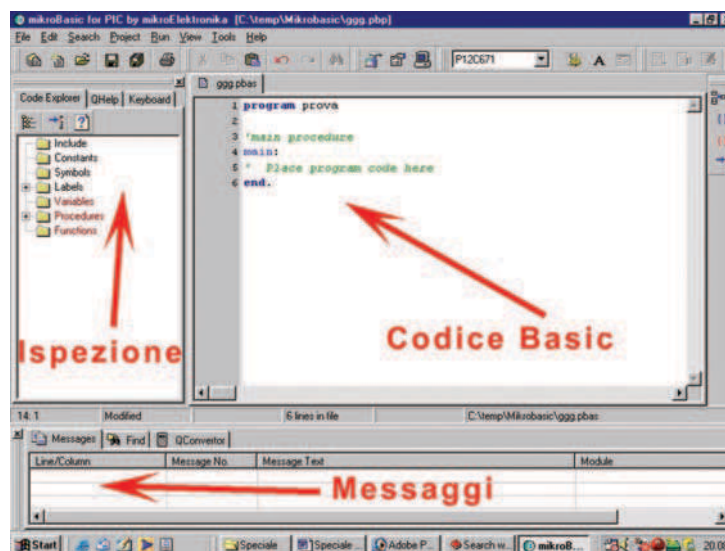
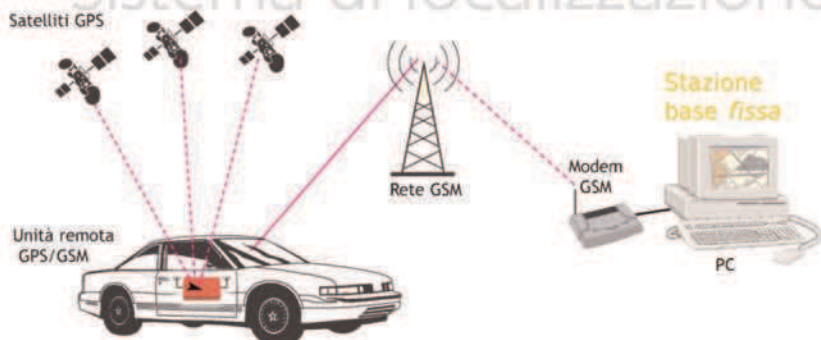



Figura 5 *Schermata dell'ambiente IDE*

Sistema di localizzazione remota GPS/GSM

Sistema di localizzazione remota GPS/GSM

Sistema di controllo a distanza GPS/GSM in grado di stabilire la posizione di un veicolo e di ascoltare quanto viene detto all'interno dello stesso.
Il sistema è composto da un'unità remota (montata sulla vettura) e da una stazione base che utilizza un PC, un'apposito software di connessione, un software cartografico con le mappe dettagliate di tutta Italia ed un modem GSM per il collegamento.
Per l'ascolto ambientale è sufficiente l'impiego di un telefono fisso o di un cellulare.

Il REM2004 comprende tutti gli elementi hardware e software necessari per realizzare una stazione base con la quale visualizzare in tempo reale la posizione di un'unità remota GSM/GPS, scaricare i dati relativi al percorso, programmare tutte le funzioni, visualizzare i dati storici, eccetera. L'unico elemento non compreso è il PC. Il software di gestione è compatibile con l'unità remota con memoria FT521K. Per la connessione all'unità remota questo sistema utilizza un modem GSM che deve essere reso attivo con l'inserimento di una SIM card valida. La SIM card non è compresa. Il set REM2004 è composto dai seguenti elementi:

- Modem GSM bibanda GM29;
- Antenna a stilo GSM bibanda con cavo di connessione;
- Alimentatore da rete per modem GM29;
- Cavo seriale DB9/DB9 per collegamento al PC;
- Software di connessione e gestione REM2004 (SFW521);
- Software di gestine cartografica Fugawi 3.0 con chiave hardware (USB);
- CD con mappe stradali di Italia, Svizzera e Austria EUSTR2).

Disponibili mappe dettagliate di tutta Europa.

REM2004 Euro 560,00

Compatta unità remota di localizzazione e ascolto ambientale che utilizza le reti GPS e GSM per rilevare la posizione del veicolo e trasmettere i dati alla stazione di controllo. Il circuito dispone inoltre di un sistema di ascolto ambientale. L'unità remota comprende anche il ricevitore GPS con antenna integrata, l'antenna GSM ed il microfono preamplificato.

Caratteristiche elettriche generali

Alimentazione 12 VDC; Assorbimento a riposo: 110 mA (GPS attivo); Assorbimento in collegamento: 380/480 mA; Memoria dati: 8.192 punti; Sensibilità microfonica max -70 dB; Dimensioni: 35 x 70 x 125 mm (esclusa antenna GPS); Sensore di movimento al gas di mercurio.

Funzionalità

Completamente teleconfigurabile; Password di accesso; Funzionamento in real time; Memorizzazione dati su remoto (8.192 punti); Tempo di polling regolabile; Sensore di movimento programmabile; Attivazione GPS programmabile; SMS di allarme gestito da sensore di movimento; Verifica tensione di batteria con gestione SMS di allarme; Ascolto ambientale configurabile da remoto;

Sezione GPS

Ricevitore GPS 12 canali con antenna attiva; Tecnologia SiRF II Low Power; Sensibilità - 170 dBW; Uscita seriale a 4800 Bps; Protocollo NMEA; 0183 V2.2; Tempo di (ri)acquisizione 38 o 8 (ri) secondi; Dimensioni 59 x 47 x 21 mm; Assorbimento inferiore a 90 mA; Tensione di alimentazione 5 V DC; Batteria di back-up interna; Temperatura di lavoro -40°C / + 80°C.

Sezione GSM

Modulo GSM/GPRS GM47; Banda 900/1800 MHz; Classe 4 (2W @ 900 MHz); Classe 1 (1W @ 1800 MHz); Alimentazione 3,6 VDC; Assorbimento a riposo 5 mA; Assorbimento in comunicazione 250/350 mA; Short Messages Service (SMS); Circuito asincrono dati non transparente fino a 9.6 kbps; Temperatura di lavoro -25°C to +55°C; Dimensioni 50 x 33 x 7.2mm; Peso 18.5g.

FT521 Euro 480,00

Tutti i prezzi sono da intendersi IVA inclusa.

Ricevitori ed antenne GPS

GPS910 Euro 98,00

Completo ricevitore GPS miniaturizzato con antenna incorporata basato su tecnologia SiRF.

- Capacità di gestire 12 satelliti contemporaneamente;
- protocollo di trasmissione tipo NMEA;
- tempo di acquisizione dei satelliti da 0,1 a 48 sec;
- tempo di rinfresco 1 sec.

Versione seriale.
Studiato per un collegamento al PC, dispone di connettore seriale a 9 poli e MiniDIN PS/2 da cui preleva l'alimentazione.

Versione USB
Dispone di un connettore standard USB da cui preleva anche l'alimentazione. Completo di software di installazione.

GPS910U Euro 110,00

RICEVITORE: SiRF II Low Power, 12 canali
SENSIBILITA': - 170 dBW
ANTENNA: attiva, incorporata
PROTOCOLLO: NMEA 0183 V2.2
PRECISIONE ORIZZ: 15 m 95° (SA off)
PRECISIONE VERT: 40 m 95% (SA off)
PRECISIONE VELOCITA': 0,1 m/sec
PRECISIONE TEMPORALE: 1 µsec
DIMENSIONI: 59 x 47 x 21 mm
ASSORBIMENTO: inferiore a 90 mA
LED: Si illumina in presenza di segnale valido all'uscita (fix)
TEMPERATURA OP: -40°C / + 80°C
UMIDITA' MASSIMA: 95%
CONTENITORE: plastico a tenuta stagna con base magnetica.

GPS901 Euro 18,50

Piccolissima ed economica antenna attiva GPS ad elevato guadagno munita di base magnetica. Può funzionare in abbinamento a qualsiasi ricevitore GPS dal quale preleva la tensione di alimentazione.

Caratteristiche elettriche: Frequenza di lavoro: 1.575,42 MHz, VSWR: 1.5 max, Guadagno: 27 dB, Alimentazione: 3 ÷ 5 VDC, Assorbimento: 22 mA, Dimensioni: 50 x 50 x 17 mm, Peso: 110 grammi, Cavo: RG174 (5 metri), Connettore: MCX 90°.

Nuova SEDE Nuovi RECAPITI
FUTURA ELETTRONICA
Via Adige, 11 -21013 Gallarate (VA)
Tel. 0331/799775 - Fax. 0331/778112 *www.futuranet.it*

Maggiori informazioni su questi prodotti e su tutti le altre apparecchiature distribuite sono disponibili sul sito *www.futuranet.it* tramite il quale è anche possibile effettuare acquisti on-line.

al nostro PIC.
Approntiamo lo schema visibile in figura 6, collegando anche un diodo led in serie ad una resistenza di limitazione alla porta RB0 del PIC.
Creiamo in mikrobasic un nuovo progetto, definendone il nome a nostro piacimento, la cartella di destinazione ed il tipo di microcontrollore utilizzato (nel nostro caso PIC16F84A).
Abilitiamo anche i check box relativi al **WDT_OFF e XT_OSC** (vedi figura 8). Nell'area del codice scriviamo il seguente programma:

```
program led
main:
  trisb.0=0  'Definisce il pin
              'RB0 come Uscita

  portb.0=1  'Pone il pin RB0 a
              'livello logico alto 1 (5V)
end.
```

Se avete scritto correttamente il codice, potete tranquillamente compilare selezionando il menu *Run* e scegliendo la voce *Compile.*
Il mikrobasic genererà un file con estensione .HEX. Sarà proprio questo il file che dovrà essere copiato all'interno del nostro PIC attraverso un programmatore hardware connesso alla porta seriale o Usb del nostro computer, come la scheda *EasyPIC 2* che potete trovare sul sito di Fare Elettronica nella sezione *shop.*
Nel secondo caso potremmo usare lo stesso Mikrobasic per effettuare tale operazione, altrimenti potremmo utilizzare il programmatore presentato nel numero 222 (Dicembre 2003) di Fare Elettronica.
Alimentiamo il nostro circuito e vedremo con sorpresa che il diodo LED collegato alla porta RB0 si illumina.
Certo, utilizzare un microcontrollore per compiere solamente un'operazione così semplice non sarebbe certamente la soluzione più economica, però ci serve per capire come un programma, tramite istruzioni ordinate e sequenziali, riesca a far cambiare il livello logico, e quindi il voltaggio, di una porta del PIC.

Esaminiamo attentamente le istruzioni del nostro primo programma. La prima linea **program** informa il compilatore che da quel punto inizia un programma. La seconda linea **main:** serve a definire un punto di riferimento chiamato label o etichetta. Più avanti spiegheremo di cosa si tratta. La terza linea **trisb.0=0** comunica al PIC che il pin RB0 sarà considerato come uscita e non come ingresso. La quarta linea **portb.0=1** invece pone a 1 logico il pin RB0 del PIC. Con un tester noteremo che tra tale piedino e la massa si misurano esattamente 5 V. Infine la quinta linea **end.** comunica appunto la fine sequenziale delle istruzioni e quindi del programma.

INGRESSO E USCITA

Bene, spegniamo per un attimo il circuito e spieghiamo dettagliatamente come avviene la

Figura 6 *Schema di prototipo con LED*

Figura 7 *La porta B in ingresso e uscita*

programmazione delle porte. Come detto in precedenza, il PIC dispone di 13 pin utilizzate per comunicare con il mondo esterno, cioè la PORTB formata da 8 bit (contrassegnati RB0, RB1, RB2, RB3, RB4, RB5, RB6, RB7) e la PORTA formata da 5 bit (contrassegnati RA0, RA1, RA2, RA3, RA4). Ognuno di questi bit e/o pin può essere configurato indipendentemente come ingresso (a cui possiamo collegare un pulsante, un interruttore, un sensore o altro) oppure come uscita (a cui possiamo collegare un led o un transistor di potenza per pilotare carichi maggiori). Per gestire al meglio queste porte si usano diversi metodi, alcuni peraltro molto semplici. Consigliamo vivamente di utilizzare sempre i registri per la configurazione di esse, vale a dire il registro **TRIS** ed il registro **PORT.**
Consideriamo per esempio la PORTB composta da 8 ingressi/uscite.
Il PIC non potrà mai sapere in anticipo se è nostra intenzione collegare su RB0 un LED verde, su RB2 un LED rosso, su RB5 un interruttore e su RB7 un pulsante: dobbiamo essere noi, poveri programmatori, a renderlo noto. Questa assegnazione di compiti è effettuata appunto grazie al registro **TRISA** (per la Porta A) e **TRISB** (per la Porta B).
Guardiamo per un attimo la figura 7. La porta B è utilizzata per illuminare alcuni diodi colorati e per accettare dall'esterno dei segnali. Facciamo anche riferimento alla figura 1. I terminali collegati ai diodi LED sono chiaramente delle **uscite**, mentre i terminali collegati a pulsanti ed interruttori sono **ingressi.** Ebbene, per comunicare al PIC questa nostra scelta, occorre assegnare al registro TRISB un opportuno valore. Ciò si può effettuare in due modi possibili:

1. Specificare per ogni pin la sua funzione di Input o Output (ed in particolare occorre dare il valore **0** se si tratta di uscita, **1** se si tratta di ingresso) con queste istruzioni:

```
trisb.0=0  'OUT
trisb.2=0  'OUT
trisb.5=1  'In
trisb.7=1  'In
```

   In questo modo abbiamo definito la funzione specifica per ogni pin. Attenzione a non porre mai a livello logico 1 un ingresso (specialmente se è direttamente cortocircuitato a massa). Il numero dopo il punto indica quale pin si deve programmare (da 0 a 7).
2. Usando solo una istruzione "cumulativa" per tutti i pin (basta sommare il numero blu della figura 7 in corrispondenza degli ingressi (valore logico 1):

```
trisb=160 '(128+32)
```

   In un'unica istruzione abbiamo inglobato tutte le funzioni della porta B. Come si nota, non è stato specificato il numero del singolo pin. Le stesse regole valgono per la PORTA A, con la sola differenza che essa ha solamente 5 pin contro gli 8 della porta B.

LA TEMPORIZZAZIONE

Il micro esegue in un secondo 1 milione di istruzioni macchina, numero di tutto rispetto per un chip così piccolo.
Ci sono però molte occasioni in cui questa velocità non serve. Un esempio concreto è rappresentato da un lampeggiatore. Se non mettessimo tra i vari lampeggi del led un' istruzione che in qualche modo ritardi gli impulsi, i lampeggi stessi avanzerebbero con una tale rapidità che l'occhio umano non si accorgerebbe di nulla.
Ecco quindi il nostro prossimo progetto: quello di realizzare un lampeggiatore a Led con cadenza di un secondo (1000 millisecondi). Utilizziamo ancora lo schema di figura 6. Creiamo un nuovo progetto, chiamandolo *lampeggio*, specifichiamo la cartella di lavoro, definiamo il PIC 16F84A quale microcontrollore e abilitiamo i flags WDT_OFF e XT_OSC. Scriviamo quindi il seguente programma:

```
program lampeggio
main:
```

```
  trisb.0=0  'RB0 come uscita
inizio:
  portb.0=1 'Accende LED
  Delay_ms(1000)'Wait 1 sec
  portb.0=0 'Spegne LED
  Delay_ms(1000)'Wait 1 sec
  goto inizio 'Ripete sempre
end.
```

In questo listato vi sono alcuni concetti nuovi da chiarire. Spieghiamo dettagliatamente cosa avviene in ogni linea di programma, partendo dalla prima:

Figura 8 Configurazione di un progetto

1. Inizia il programma dal nome *lampeggio*.
2. Definisce la label o punto di riferimento (main).
3. Definisce il pin RB0 come uscita.
4. Definisce la label inizio.
5. Mette a 1 logico il pin RB0.
6. Aspetta 1 secondo (1000 ms).
7. Mette a 0 logico il pin RB0.
8. Aspetta 1 secondo (1000 ms).
9. Ritorna a eseguire le istruzioni dalla label **inizio.**
10. Fine (non viene praticamente mai eseguita).

Compiliamo il programma e con il programmatore built-in o con IC-Prog (o Pic Prog) carichiamo il file *lampeggio.hex* sulla memoria del PIC.
Collegando l'alimentazione al nostro circuito vedremo che il diodo LED lampeggia esattamente alla cadenza di un secondo.

Come vedete, il metodo migliore per imparare l'utilizzo del PIC è proprio quello di provare e provare ancora, magari sbagliando. Male che vada mandiamo in fumo il microcontrollore ma, per il suo costo estremamente contenuto, vale proprio la pena di tentare. Dovete sapere anzi che i processori utilizzati circa 50 anni fa costavano milioni di dollari, erano grandi quanto un appartamento e a livello di prestazioni non si avvicinavano neppure lontanamente al nostro 16F84. Vedrete col tempo che le routines di temporizzazione saranno sicuramente le più utilizzate: in un progetto occorre sempre conciliare quelli che sono i tempi macchina ai ritmi più lenti dell'uomo.
Il nostro compilatore dispone di due utili istruzioni per mettere il PIC in pausa per un certo tempo: la prima è *delay_ms(t)* che ferma l'esecuzione dei comandi per t millesimi di secondo; la seconda è *delay_us(t)* che ferma l'esecuzione dei comandi per t milionesimi di secondo.

Figura 9 Schema con due diodi led

AUMENTIAMO IL NUMERO DELLE USCITE

Inseriamo adesso un secondo diodo LED al circuito, in modo da averne uno rosso su RB0 e uno verde su RB1. Tali porte vanno quindi considerate come delle uscite (output). Realizziamo lo schema presente in figura 9.

Il nostro scopo stavolta è quello di far lampeggiare alternativamente i due LED, al ritmo di mezzo secondo. Le cose cominciano pian piano a complicarsi. Per gestire meglio gli eventi proviamo a tracciare uno schema a blocchi. Creiamo allora la seguente flow-chart o diagramma di flusso, che spiega passo per passo cosa avviene nel PIC in maniera schematica, come da figura 10.

Creiamo con il compilatore un nuovo progetto, specificando sempre gli stessi parametri visti in precedenza, e scriviamo questo codice:

```
program lampeggio
main:
  trisb.0=0  'RB0 come uscita
  trisb.1=0  'RB1 come uscita
inizio:
  portb.0=1 'Accende LED 1
  portb.1=0 'Spegne LED 2
  Delay_ms(500)'Wait 0,5 sec
  portb.0=0 'Spegne LED 1
  portb.1=1 'Accende LED 2
  Delay_ms(500)'Wait 0,5 sec
  goto inizio 'Ripete sempre
end.
```

Dando tensione al circuito vedremo i due diodi LED lampeggiare alla frequenza di mezzo secondo.

DEDUZIONI

Tutto ciò che abbiamo fatto sino ad ora ci porta a trarre le seguenti conclusioni:

- Un software può essere scritto in tanti modi diversi, anche secondo il proprio stile di ragionamento.
- Il PIC segue ciecamente quanto programmato, quindi se certi risultati sono errati, il problema è da imputare all'uomo e non alla macchina.
- Non esiste problema che non può essere risolto con un linguaggio di programmazione.
- Scrivere in Basic è molto semplice, ma scrivere in Assembler dà più conoscenze dell' hardware utilizzato, anche sacrificando chiarezza di codice e tempi di sviluppo.
- La velocità di esecuzione del software scritto dipende anche dal quarzo esterno utilizzato.
- Le uscite del PIC possono solo "pilotare" piccolissimi carichi. Per ottenere potenze maggiori è necessario utilizzare alcuni componenti per amplificare il segnale, ma ciò sarà approfondito successivamente.
- In un programma occorre prevedere TUTTE le possibili evenienze, anche quelle più remote.
- Abituatevi a inserire molti commenti nel codice scritto: commentare vuol dire scrivere appunti e annotazioni nel codice stesso che spiegano le funzioni svolte. Ciò vuol dire massima leggibilità nostra o di altri programmatori, anche in fase di debug e modifiche varie.
- Le uscite o ingressi del PIC sono "solamente" 13. È possibile però con accorgimenti particolari aumentare questo numero.

IL NOSTRO PRIMO SEMAFORO

Bene, siamo già maturi per creare il prossimo progetto, quello di un semaforo stradale. Dobbiamo innanzitutto analizzare cosa avviene durante un ciclo semaforico completo:

1. Si accende il verde. Le automobili passano.
2. Il verde continua a restare acceso per 30 secondi.
3. Mentre il verde è acceso, si illumina il giallo per 5 secondi.
4. Si accende il rosso. Il verde e il giallo si spengono.
5. Il rosso resta acceso per 30 secondi.
6. Si spegne il rosso ed il ciclo ricomincia da capo.

Ricordate sempre che l'analisi del problema deve essere studiata in maniera perfetta e soprattutto

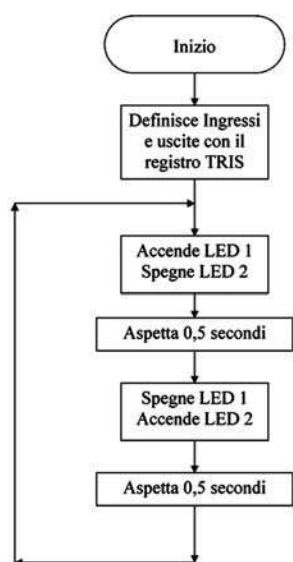



Figura 10 *Flow chart per il lampeggio alternato di due led*

deve essere esente da errori di valutazione. In definitiva il PIC è solo il "braccio" che esegue i comandi, e la "mente" è rappresentata dal programmatore.

Assembliamo lo schema visibile in figura 11, che prevede il LED rosso su RB0, il LED giallo su RB1 ed il LED verde su RB2.

Adesso scriviamo il codice Basic, tenendo sott'occhio l'analisi che abbiamo prima sviluppato:

```
program Semaforo

main:
  'Definiamo RB0 RB1 RB2
  'come output

  trisb.0=0 'RB0 OUT
  trisb.1=0 'RB1 OUT
  trisb.2=0 'RB2 OUT

  'Spegniamo inizialmente
  'tutte le tre porte
  'per sicurezza e per
  'evitare eventuali
  'shock iniziali alle
  'periferiche collegate
  portb.0=0 'RB0 OFF
  portb.1=0 'RB1 OFF
  portb.2=0 'RB2 OFF

  inizio:   'label del ciclo

  portb.2=1 'Verde ON
  Delay_ms(30000) 'Wait 30 sec.

  portb.1=1 'Anche giallo ON
  Delay_ms(5000) 'Wait 5 sec.

  portb.2=0 'Verde OFF
  portb.1=0 'Giallo OFF

  portb.0=1 'Rosso ON
  Delay_ms(30000) 'Wait 30 sec.
  portb.0=0 'Rosso OFF

  goto inizio 'Ricomincia tutto

end.
```

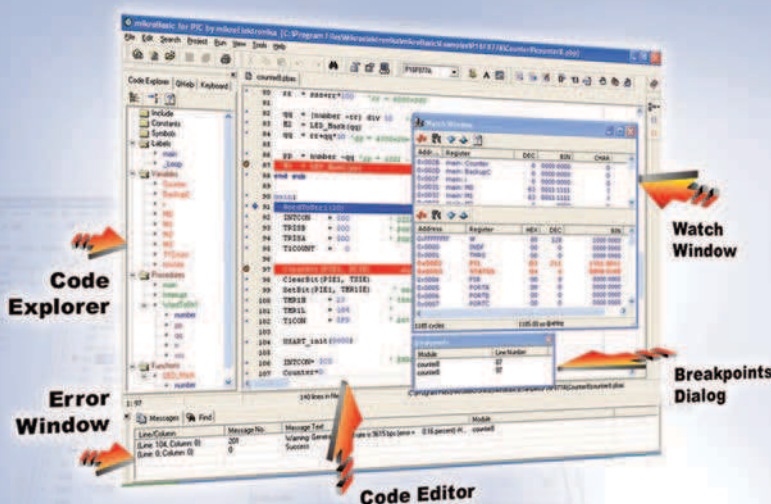
Compiliamo il programma, carichiamolo sul nostro PIC e vedremo con grande soddisfazione che il semaforo funziona egregiamente, rispettando i tempi programmati ed eseguendo infinite volte l'intero ciclo con incredibile affidabilità.

Prima dell'avvento dei microcontrollori tali progetti erano realizzabili solo utilizzando decine di circuiti integrati, flip-flop, contatori, convertitori, che rendevano complicato uno schema semplice come il nostro. C'è da dire anche che in passato, una modifica al progetto avrebbe implicato la completa ricostruzione dei prototipi mentre con il PIC basta modificare solamente il *programma* per rendere subito efficace la variazione. Senza intervenire minimamente sul circuito! Come detto prima, lo schema base può già da solo costituire una valida *Pic Test Board,* ossia una scheda per effettuare tutte le prove possibili ed immaginabili. Basta dotare i terminali del microcontrollore di alcuni morsetti per permettere il collegamento provvisorio a eventuali periferiche esterne, come Led, lampade, pulsanti e interruttori. In figura 12 potete notare un esempio di tutto questo. Per la sua notevole semplicità, la Test Board può essere approntata anche su basetta mille fori.

L'ALGORITMO

Dopo questi esempi si nota subito una grande differenza tra come "pensa" la macchina e come pensa l'uomo. La macchina ha necessità di avere specificate, passo dopo passo, le varie operazioni da eseguire, e, seppur semplici, que-

Figura 11 *Schema elettrico del Semaforo*

da confronti e deduzioni complesse che, forse, la macchina non riuscirà mai ad emulare.Il programmatore deve quindi risolvere un determinato problema sostituendosi alla macchina e pensando come essa. Deve trovare il metodo di risoluzione, codificarlo e insegnarlo al processore. Se egli non ha chiari questi concetti, difficilmente il software sviluppato funzionerà bene. Introduciamo allora un argomento importantissimo, che segna la teoria del software e che traccia i fondamentali passaggi

ste devono essere spiegate in modo molto dettagliato. L'uomo invece ha un approccio totalmente differente: il suo pensiero neurale deriva

del suo sviluppo: **l'Algoritmo**. Esso si può definire come una sequenza di compiti successivi eseguibili in modo meccanico per risolvere un

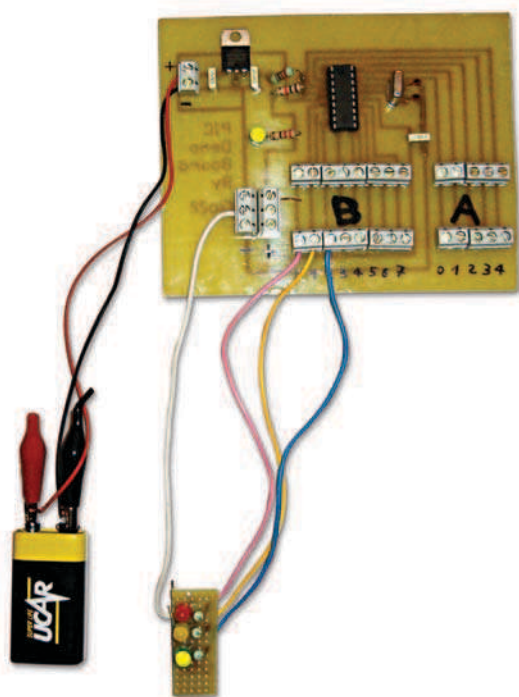



Figura 12 *La Pic Test Board con il semaforo*

problema. Per facilitare la comprensione e la schematizzazione, esso può venir rappresentato tramite un *diagramma di flusso*, ossia uno schema a blocchi, dove ogni forma rappresenta un'azione che il calcolatore deve eseguire. Nel corso degli eventi utilizzeremo molto i diagrammi di flusso (o Flow chart), ma vedrete che con la pratica se ne può anche farne a meno.

PRIMO APPROCCIO AL CICLO ITERATIVO

I programmi fin'ora visti sono molto semplici e fanno uso di poche risorse software e hardware. Quando le esigenze operative aumentano, anche il linguaggio di programmazione ci viene sempre più d'aiuto, mettendoci a disposizione ulteriori mezzi di sviluppo. Supponiamo di risolvere il seguente problema: si hanno a disposizione due lampadine (o LED), una rossa e una verde. Inizialmente la verde è spenta. La rossa deve invece lampeggiare per 5 (cinque) volte alla frequenza di 1 Hz (una volta al secondo). Al termine del lampeggio si deve accendere permanentemente il LED verde. Realizziamo nuovamente lo schema proposto in figura 9. Con le conoscenze attuali, il programma (peraltro *funzionante* ma *NON funzionale*) è il seguente:

```
program Lampeggio2

main:
   'definisce funzione pin
   trisb.0=0 'RB0 out
   trisb.1=0 'RB1 out

   'spegne tutto
   portb.0=0 'RB0 OFF
   portb.1=0 'RB1 OFF

   '--Primo lampeggio--
   portb.0=1 'rosso ON
   Delay_ms(500)
   portb.0=0 'rosso OFF
   Delay_ms(500)

   '--Secondo lampeggio--
   portb.0=1 'rosso ON
   Delay_ms(500)
   portb.0=0 'rosso OFF
   Delay_ms(500)

   '--Terzo lampeggio--
   portb.0=1 'rosso ON
   Delay_ms(500)
   portb.0=0 'rosso OFF
   Delay_ms(500)

   '--Quarto lampeggio--
   portb.0=1 'rosso ON
   Delay_ms(500)
   portb.0=0 'rosso OFF
   Delay_ms(500)

   '--Quinto lampeggio--
   portb.0=1 'rosso ON
   Delay_ms(500)
   portb.0=0 'rosso OFF
   Delay_ms(500)

   '--Finalmente accende verde
   portb.1=1 'Verde ON

end.
```

Compilate il programma e dopo il suo scaricamento sul PIC vedrete cinque lampeggi del LED rosso e poi la definitiva accensione del LED verde. Questa soluzione potrebbe andare bene, ma se per ipotesi il numero dei lampeggi dovesse aumentare da *cinque* a *cento* inizierebbero i problemi. Dovremmo riscrivere per altre 95 volte il blocco di codice relativo alla sequenza; ne verrebbe un programma molto lungo, sia per quanto riguarda il codice che per l'eseguibile. Probabilmente il PIC stesso lo rifiuterebbe.
Ecco allora che il linguaggio Basic ci mette a disposizione una funzione estremamente potente, che ci permette di ripetere *tante volte* un insieme di istruzioni. Tale soluzione prende il nome di **ciclo**. Ritornando al nostro problema, guardiamo bene il nostro listato: vediamo che c'è una parte di codice che, ripetendosi per cinque volte, è sempre la stessa.
Ci riferiamo all'accensione e allo spegnimento del diodo rosso, con i relativi tempi di attesa. Modifichiamo il sorgente con l'uso dei cicli, poi commenteremo:

```
program Lampeggio2

dim k as byte

main:

  trisb.0=0 'RB0 out
  trisb.1=0 'RB1 out

  portb.0=0 'RB0 OFF
  portb.1=0 'RB1 OFF

  for k=1 to 5
    portb.0=1 'rosso ON
    Delay_ms(500)
    portb.0=0 'rosso OFF
    Delay_ms(500)
  next k

  '--Finalmente accende verde
  portb.1=1 'Verde ON

end.
```

Compilando ed eseguendo sul PIC il programma noteremo che il risultato finale è lo stesso. Guardando però il sorgente esso è passato dalle iniziali 36 righe di codice a sole 16 righe, ed anche il file eseguibile risulta molto più snello e *leggero*.
Quali sono le novità? Innanzitutto non ci sono più i 5 blocchi di codice che servivano ad accendere e spegnere il LED rosso, ma ne è rimasto uno solo. Tale blocco è "chiuso" all'interno di un *ciclo di ripetizione*, cioè una sorta di scatola che viene ripetuta per un numero indefinito. Inizia con il comando **for k=1 to 5** e termina con il comando **next k**.
Letteralmente vuol dire "considera per K tutti i valori numerici da 1 a 5"; in altre parole, tutto ciò che è racchiuso tra la parola chiave **for** e la parola **next** sarà ripetuto per cinque volte.
Dato che k rappresenta un valore numerico (avremmo potuto usare qualsiasi altra lettera), lo abbiamo inserito all'inizio del programma usando la definizione di variabile (dim k as byte). Un ulteriore esempio chiarirà meglio le idee. Riferendoci stavolta allo schema di cui in figura 11, abbiamo la necessità di far lampeggiare alla cadenza di 2 Hz il LED rosso per 5 volte, il LED giallo per 10 volte ed il LED verde per 7 volte. Studiamo attentamente il listato:

```
program Lampeggio3

dim ripeti as byte

main:
  trisb.0=0 'RB0 Out
  trisb.1=0 'RB1 Out
  trisb.2=0 'RB2 Out
  '--Spegne TUTTA la PORTB
  portb=0 '00000000 (8 bit a zero)

inizio:
  '-Ciclo per diodo ROSSO su RB0-
  for ripeti=1 to 5
    portb.0=1 'RB0 ON
    Delay_ms(250)
    portb.0=0 'RB0 OFF
```

```
    Delay_ms(250)
  next ripeti

  '-Ciclo per diodo GIALLO su RB1-
  for ripeti=1 to 10
    portb.1=1 'RB1 ON
    Delay_ms(250)
    portb.1=0 'RB1 OFF
    Delay_ms(250)
  next ripeti

  '-Ciclo per diodo VERDE su RB2-
  for ripeti=1 to 7
    portb.2=1 'RB2 ON
    Delay_ms(250)
    portb.2=0 'RB2 OFF
    Delay_ms(250)
  next ripeti

  goto inizio  'Ripeti TUTTO

end.
```

Stavolta abbiamo utilizzato la variabile **ripeti**, che ha il compito di contare il numero lampeggi dei vari LED. Il nome della variabile può essere qualsiasi, purché non infranga le regole del mikrobasic.
Dedicheremo il prossimo capitolo allo studio delle variabili.
Grazie ai cicli, il codice risulta essere *enormemente* più corto e ottimizzato, più leggibile e facilmente modificabile, in quanto la parte da reiterare è inserita tra i comandi FOR e NEXT.
Riprendiamo un po' di fiato provando un piccolo esercizio.
Colleghiamo a TUTTI i pin della PORTB otto diodi LED (uno per ogni pin) con la relativa resistenza di limitazione da 470 ohm.
Dopo la definizione di output della PORTB, occorre programmare i seguenti comportamenti dei LED:

1. Spegnimento iniziale di tutti i LED.
2. Si accende il primo LED per 3 secondi.
3. Si accende il secondo LED per 3 secondi (il primo resta acceso).
4. Si spengono i due LED restando spenti per 2 secondi.
5. Si accendono i LED 3, 4, 5 per 5 secondi.
6. Si spengono quindi tali LED e si accendono i LED 6, 7, 8 per 5 secondi.
7. Si spengono tutti i LED.
8. I LED 1, 3, 5, 7 lampeggiano assieme (ON-OFF) alla cadenza di 1 secondo, per 10 volte.
9. Si accendono TUTTI i LED per 10 secondi(il PIC inizia a sudare …).
10. Finalmente tutti i LED si spengono (con una sola istruzione).

Mamma mia, stiamo iniziando a fare sul serio. Potete inviare le risposte alla e-mail di intestazione dell'articolo, nonché quesiti su eventuali problemi e malfunzionamenti del circuito.

Avete mai visto le centraline di allarmi e antifurto? Se avete notato, all'accensione del circuito, vi sono alcuni LED che lampeggiano per un numero definito di volte, facendo intendere che l'antifurto è in fase di "riscaldamento" o di preparazione al funzionamento. Ebbene, questi cicli di lampeggio sono proprio effettuati tramite microcontrollore, che coordina pazientemente tutte queste operazioni.
Pian piano vedrete che alcuni concetti saranno sempre più chiari: basta solamente molta costanza e studio.
Un consiglio che diamo è quello di rileggere tante volte l'articolo e di studiarlo, magari sforzandovi di afferrare un concetto inizialmente non troppo chiaro.

UN PICCOLO GIOCO DI LUCI

Ricordate il telefilm Supercar? Il protagonista aveva sulla sua vettura un fila di lucine che si illuminavano da destra verso sinistra e viceversa. Ebbene, simuliamo questo gioco di luci con il microcontrollore.
Per il nostro scopo proviamo tre sorgenti Basic, tutti funzionanti, ma uno più "perfezionato" dell'altro.

Realizziamo per questo progetto lo schema che vedete in figura 13. Su tutte le otto uscite della PORTB sono collegati altrettanti diodi LED con la relativa resistenza di limitazione.

Il programma deve illuminare il primo LED; dopo 100 ms. (un decimo di secondo) deve accendere il secondo LED, il terzo e, così via, sino all'ottavo LED, poi il settimo, il sesto sino al secondo LED.
Il ciclo così continua all'infinito.
In figura 14 vediamo l'ordine di accensione dei LED nelle 14 sequenze cicliche (ogni sequenza è numerata progressivamente).

Primo Metodo

Il primo metodo prevede l'accensione di un *singolo* LED e lo spegnimento contemporaneo del precedente.
In questa maniera il codice Basic risultante potrebbe essere molto lungo:

```
program SuperCar
main:
  trisb=0 'Definisce la
          'PORTB tutta in
          'Uscita (%00000000)

portb=0 'Spegniamo tutti i Pin
        'Equivale a %00000000

ciclo:
  portb.0=1 'Led 1 ON
  delay_ms(100)

  portb.0=0 'Led 1 OFF
  portb.1=1 'Led 2 ON
  delay_ms(100)

  portb.1=0 'Led 2 OFF
  portb.2=1 'Led 3 ON
  delay_ms(100)

  portb.2=0 'Led 3 OFF
  portb.3=1 'Led 4 ON
  delay_ms(100)

  portb.3=0 'Led 4 OFF
  portb.4=1 'Led 5 ON
  delay_ms(100)

  portb.4=0 'Led 5 OFF
```

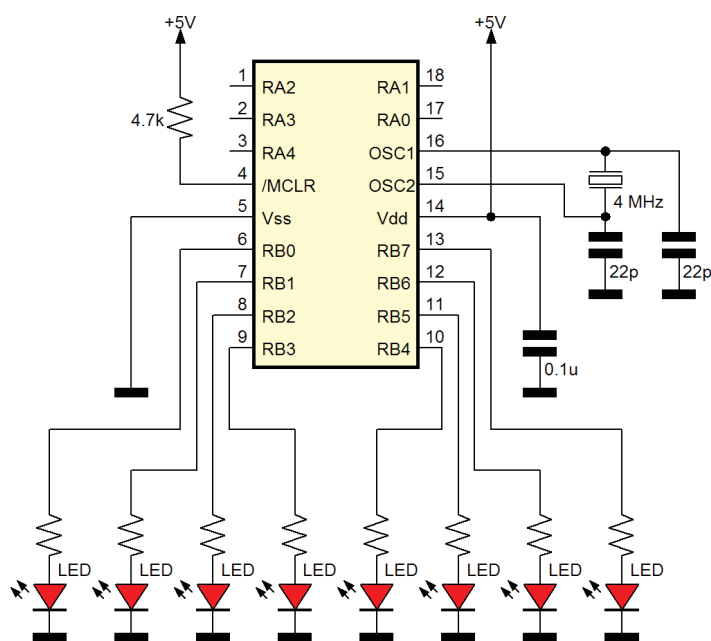



Figura 13 *Schema con 8 LED su PORTB*

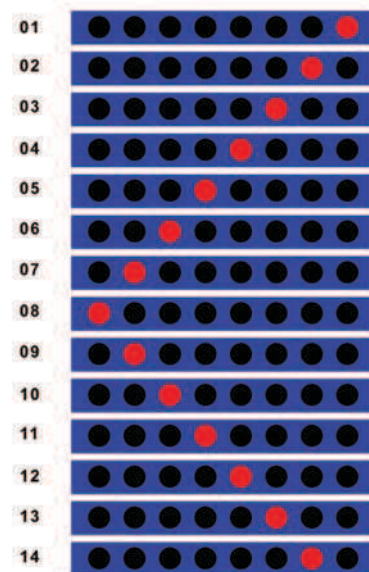



Figura 14 *La sequenza di accensione*

```
     portb.5=1 'Led 6 ON
     delay_ms(100)

     portb.5=0 'Led 6 OFF
     portb.6=1 'Led 7 ON
     delay_ms(100)

     portb.6=0 'Led 7 OFF
     portb.7=1 'Led 8 ON
     delay_ms(100)

     portb.7=0 'Led 8 OFF
     portb.6=1 'Led 7 ON
     delay_ms(100)

     portb.6=0 'Led 7 OFF
     portb.5=1 'Led 6 ON
     delay_ms(100)

     portb.5=0 'Led 6 OFF
     portb.4=1 'Led 5 ON
     delay_ms(100)

     portb.4=0 'Led 5 OFF
     portb.3=1 'Led 4 ON
     delay_ms(100)

     portb.3=0 'Led 4 OFF
     portb.2=1 'Led 3 ON
     delay_ms(100)

     portb.2=0 'Led 3 OFF
     portb.1=1 'Led 2 ON
     delay_ms(100)

     portb.1=0 'Led 2 OFF

     goto ciclo

end.
```

Secondo Metodo

Con il secondo metodo, paragoniamo gli 8 LED della PORTB a 8 bit di un byte. Pertanto, se i vari pin della porta rappresentano il numero binario della porta stessa, per abilitare le varie uscite dovremo fornire l'esatto valore corrispondente, secondo questa tabella di sequenza (il simbolo 1 indica il Led acceso mentre il simbolo 0 indica il Led spento):

01) 00000001 = 1
02) 00000010 = 2
03) 00000100 = 4
04) 00001000 = 8
05) 00010000 = 16
06) 00100000 = 32
07) 01000000 = 64
08) 10000000 = 128
09) 01000000 = 64
10) 00100000 = 32
11) 00010000 = 16
12) 00001000 = 8
13) 00000100 = 4
14) 00000010 = 2

Bene, adesso che disponiamo dei vari "pesi" numerici, possiamo scrivere un codice sicuramente più corto e più compatto:

```
program SuperCar2

main:
  trisb=0 'Definisce la
          'PORTB tutta in
          'Uscita (%00000000)

  portb=0 'Spegniamo tutti i Pin
          'Equivale a %00000000

  ciclo:

    portb=1
    delay_ms(100)
    portb=2
    delay_ms(100)
    portb=4
    delay_ms(100)
    portb=8
    delay_ms(100)
    portb=16
    delay_ms(100)
    portb=32
    delay_ms(100)
```

```
    portb=64
    delay_ms(100)
    portb=128
    delay_ms(100)
    portb=64
    delay_ms(100)
    portb=32
    delay_ms(100)
    portb=16
    delay_ms(100)
    portb=8
    delay_ms(100)
    portb=4
    delay_ms(100)
    portb=2
    delay_ms(100)

    goto ciclo

end.
```

Terzo Metodo

Il terzo metodo risulta essere il più corto ed ottimizzato di tutti. Utilizza variabili ed array, un concetto che approfondiremo nella prossima sezione. Ecco il listato, poi passiamo ai commenti:

```
program SuperCar3

const sequenza as
byte[14]=(1,2,4,8,16,32,64,128,64,32,16,8,4,
2)
dim k as byte

main:
  trisb=0 'PORTB OUTPUT
  portb=0 'Spegne portb
  ciclo:
    for k=0 to 13
      portb=sequenza[k]
      Delay_ms(100)
    next k
    goto ciclo
end.
```

Fantastico, lo stesso programma è divenuto cortissimo, sia in termini di codice sorgente, sia in termini di spazio eseguibile. Con l'esperienza si riuscirà a scrivere programmi sempre più ottimizzati. Si potrebbe fare ancora di meglio: comunque meglio non esagerare... almeno per il momento.

In breve diciamo che il metodo utilizza un nuovo concetto: l'uso di *variabili* e di *array* che approfondiremo nel prossimo articolo.

Cosa abbiamo fatto? In pratica abbiamo memorizzato nella memoria del PIC i 14 passi corrispondenti alla combinazione di LED che vogliamo illuminare. Il programma riprodurrà tali valori sugli 8 LED con tempo di attesa di 100 millisecondi.

Eseguite stavolta questo programma:

```
program SuperCar4

const sequenza as
byte[16]=(1,128,2,64,4,32,8,16,0,129,0,129,0
,255,0,255)
dim k as byte

main:
  trisb=0 'PORTB OUTPUT
  portb=0 'Spegne portb
  ciclo:
    for k=0 to 15
      portb=sequenza[k]
      Delay_ms(100)
    next k
    goto ciclo
end.
```

Il programma è lo stesso del precedente. Cambiano solo le sequenze delle luci da accendere. Provate il listato ed otterrete un simpatico gioco di luci.

CONCLUSIONI

Nella prossima lezione approfondiremo l'utilizzo delle *variabili* e delle *costanti* in Basic. Vedremo come gestire gli *array* e come interagire con il microcontrollore attraverso i suoi ingressi.

Naturalmente realizzeremo nuove ed interessanti esercitazioni.

gli appuntamenti

ALCUNE DELLE DATE INDICATE POTREBBERO SUBIRE VARIAZIONI

02-03 Aprile 2005

EXPO ELETTRONICA ERBA — Erba (CO)

EXPO ELETTRONICA (ex ABC dell'Elettronica) ad Erba (Como) si svolge due volte all'anno, in primavera ed in autunno. Oltre alle merceologie "tradizionali" proposte da questo tipo di manifestazioni, quali computer, elettronica in genere, radiantismo, telefonia, surplus... nonché radio d'epoca, dischi e CD da collezione.
Inoltre propone, in primavera, il Salone dell'Astronomia e Photo Cine video, con macchine ed attrezzature per la fotografia , mentre in autunno i protagonisti sono i radioamatori grazie al CB Day, a loro dedicato.
Certamente non mancano i buoni motivi per andare a dare un'occhiata; ci saranno buoni affari sia per gli esperti sia per i neofiti!

LUOGO: Lario Fiere - Erba (CO)
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: Blu Nautilus (www.blunautilus.it Tel 0541-53294)
INGRESSO: € 7,50 - ridotto € 6,50

09-10 Aprile 2005

TECNO ACQUI — Acqui Terme (AL)

Terza edizione di TECNO ACQUI, Mostra mercato di materiale radioamatoriale, informatico, satellitare e telefonico, nuovo ed usato.

LUOGO: Stabilimento Ex Lama Italia - Acqui Terme (AL)
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: www.tecnoacqui.it
INGRESSO: € 2,50

09-10 Aprile 2005

MOSTRA EMPOLESE DELLA RADIANTISTICA — Empoli (FI)

Terza edizione di TECNO ACQUI, Mostra mercato di materiale radioamatoriale, informatico, satellitare e telefonico, nuovo ed usato.

LUOGO: Palazzo delle esposizioni - Empoli (FI)
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: Studio Fulcro (www.studio-fulcro.it Tel 010-561111)
INGRESSO: n.p.

16-17 Aprile 2005

FIERA MERCATO DELL'ELETTRONICA — Silvi Marina (TE)

Terza edizione di TECNO ACQUI, Mostra mercato di materiale radioamatoriale, informatico, satellitare e telefonico, nuovo ed usato.

LUOGO: Fiera Adriatica – Silvi Marina (TE)
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: Studio di Antonino Camillo (Tel 085-4215840)
INGRESSO: n.p.

Per sapere a quali appuntamenti sarà presente FareELETTRONICA visita la pagina: www.farelettronica.com/fiere

21 Aprile 2005

M2M FORUM

S. Donato Milanese (MI)

L'edizione primaverile dei Wireless Forum offrirà diverse conferenze in parallelo, che tratteranno i temi più innovativi del mondo fisso e mobile, affiancate da una vasta area espositiva, pensata per soddisfare la necessità pratica degli operatori del settore di aggiornarsi su singoli prodotti, cercando di capire come utilizzare al meglio le nuove tecnologie offerte dai costruttori. In questa edizione festeggeremo la nascita del mondo "machine to machine" attraverso una galleria di prodotti che i fornitori più importanti hanno presentato in questo decennio. Tutti i prodotti saranno visibili all'interno di una bacheca allestita nell'area espositiva in occasione dell'evento. Il tutto sarà poi raccolto in un report in lingua inglese che verrà distribuito a tutti i partecipanti e allegato al numero della rivista Wireless relativo all'evento. Durante la conferenza verranno affrontati diversi argomenti sia di ampio respiro culturale sia marketing-oriented, e non mancheranno le presentazioni di casi concreti.
Temi chiave: M2M nelle nuove public utilities - Soluzioni M2M per la sicurezza di beni e persone - Novità e soluzioni per il building management - Programmazione dei moduli M2M.

LUOGO: Crowne Plaza Milan Linate – San Donato Milanese (MI)
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: Wireless (www.wirelessforum.it - Tel 02-48100306)
INGRESSO: Gratuito (registrazione necessaria)

21-23 Aprile 2005

FORTRONIC 2005

Padova (PD)

Fortronic - Electronics Forum costituisce fin dal 1983 l'impegno di **Assodel** nell'ambito dell'elettronica, e rappresenta, oggi, l'unica manifestazione nazionale dedicata all'elettronica industriale che si tiene negli anni dispari. Intervenire sulla supply chain significa saper proporre soluzioni con la competenza di una visione globale, in grado di mettere a fuoco ogni livello industriale che caratterizza le fasi di sviluppo di un prodotto o di un'apparecchiatura fino al suo termine. In quest'ottica la prossima edizione di Fortronic, si articolerà in tre giornate, ognuna dedicata a una specifica tematica, ovvero innovazione, produzione e distribuzione, con l'obiettivo di riuscire a coniugare gli interessi di tutti gli attori della "electronic community".
I settori merceologici: componenti, sensori & strumentazione, progettazione, produzione, outsourcing, servizi, automazione.

LUOGO: Fiera di Padova
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: Assodel (www.fortronic.it Tel 02-210111230)
INGRESSO: Gratuito (registrazione necessaria)

23-25 Aprile 2005

FIERA DEL RADIOAMATORE — Pordenone (PD)

Una fiera che riunisce le ultime invenzioni nel campo della radiantistica, dell'elettronica e dell'informatica, una vetrina per conoscere i progressi della tecnica e aggiornarsi sui nuovi sistemi di radiodiffusione.
Settori merceologici: apparecchi radiotrasmittenti, componenti e ricambi, attrezzature e accessori per la radiantistica, informatica, personal computer, Hi-Fi e video, elettronica di consumo, home theatre, telefonia e personal communication, Internet e servizi sulla rete, manuali, editoria specializzata.

LUOGO: Quartiere Fieristico – Pordenone
ORARI: dalle 9:00 alle 18:30
ORGANIZZATORE: Pordenone Fiere (www.fierapordenone.it Tel 043-4232111)
INGRESSO: n.p.

30 Aprile - 1 Maggio 2005

XX MOSTRA MERCATO NAZIONALE — Civitanova Marche (MC)

Mostra Mercato Nazionale della Radiantistica Elettronica: materiale radiantistico per C.B. e radioamatori, apparecchiature per telecomunicazioni, surplus, telefonia, computers, antenne e parabole per radioamatori e Tv Sat, radio d'epoca, editoria specializzata.
Salone del collezionismo: minerali, gemmologia, fossili e conchiglie, filatelia e numismatica, oggetti artigianali da collezione.
Disco: mostra mercato del disco usato in vinile e CD da collezione.

LUOGO: Quartiere Fieristico Civitanova Marche (MC)
ORARI: dalle 9:00 alle 19:30
ORGANIZZATORE: ERF (www.erf.it - Tel 0733-780815)
INGRESSO: n.p.

07-08 Maggio 2005

GRANDE FIERA DELL'ELETTRONICA

Quello di Forlì è uno degli appuntamenti "classici" per gli appassionati di mostre mercato, una "full immersion" nel mondo dell'elettronica che attira specialisti e curiosi che vogliano entrare nell'Era Digitale senza spendere una follia. Il tema generale che accomuna i principali "sottoinsiemi" della fiera è quello della comunicazione: trasmissione, ricezione, archiviazione e protezione dei dati.
A Forlì troverete computer, programmi, periferiche, accessori, telefonia fissa e mobile, sistemi di sicurezza, strumenti di misurazione, smart card, decoder, antenne e parabole, schede, apparecchiature per radioamatori, CB, ma anche radio d'epoca, antesignane dei moderni mezzi di comunicazione, pezzi rari, valvole e ricambi.

LUOGO: Fiera di Forlì – FORLI'
ORARI: dalle 9:00 alle 18:00
ORGANIZZATORE: Blu Nautilus (www.blunautilus.it Tel 0541-53294)
INGRESSO: € 7,50 - ridotto € 6,50

I Radar ad apertura

Negli ultimi decenni abbiamo assistito ad un forte impulso verso l'utilizzo delle tecnologie satellitari.
I satelliti sono utilizzati per diversi scopi che spaziano dalle telecomunicazioni, all'osservazione dei pianeti, fino alle sperimentazioni in assenza di gravità. In questo articolo dopo una descrizione generale sui satelliti ci si concentra sui satelliti con a bordo sensori radar noti come Radar ad apertura sintetica. Tali sensori sono impiegati per ricavare mappe topografiche delle aree osservate o anche per monitorare aree soggette a rischi ambientali quali frane, terremoti ed eruzioni vulcaniche.

INTRODUZIONE GENERALE SUI SATELLITI

Che cosa è un satellite?

Un satellite nel senso più ampio del termine è un qualsiasi oggetto che orbita intorno ad un corpo celeste.
In particolare si parla di *satelliti naturali* quando l'oggetto orbitante è un corpo celeste, mentre si parla di *satelliti artificiali* quando l'oggetto orbitante è stato costruito dall'uomo. La Luna per esempio è un satellite naturale poiché è un corpo celeste che orbita intorno alla Terra, mentre i satelliti METEOSAT, che come è noto sono utilizzati per le previsioni meteorologiche, costituiscono un esempio di satelliti artificiali.

Perché si utilizzano i satelliti?

I satelliti artificiali possono essere suddivisi in diverse categorie sulla base delle funzioni per cui essi sono stati progettati:

- *Satelliti per telecomunicazioni*: tali satelliti sono utilizzati per trasmettere informazione (segnali televisivi, radiofonici, telefonici o dati in genere) da un punto ad un altro della superficie terrestre o anche da un punto della superficie terrestre ad un altro pianeta (per esempio Marte). I satelliti per telecomunicazioni si propongono come alternativa alle telecomunicazioni basate su sistemi puramente terrestri.
  Grazie al loro particolare posizionamento, i satelliti per telecomunicazioni, riescono a coprire aree altrimenti impossibili da raggiungere con le convenzionali infrastrutture terrestri, ed è per questo che oggi si riesce a comunicare facilmente da una nave in pieno oceano o da luoghi nei quali fenomeni naturali violenti (terremoti, cicloni) hanno distrutto i canali di telecomunicazione terrestre. Oltre a ciò i satelliti hanno il vantaggio di riuscire a coprire con una singola antenna vaste aree, riuscendo così a servire anche popolazioni che per ragioni economiche non potrebbero disporre di un'adeguata rete di telecomunicazioni terrestri.

di Raffaele Nutricato
rnutric@tin.it

sintetica

- *Satelliti per telerilevamento*: sono utilizzati per monitorare la superficie terrestre, gli oceani e anche l'atmosfera (satelliti meteorologici). Tali satelliti sono utilizzati come integrazione alle convenzionali misure effettuate a terra, per il monitoraggio di eventi naturali quali frane, terremoti, incendi e fenomeni atmosferici. Il vantaggio nell'utilizzo di piattaforme satellitari è dato innanzitutto dalla loro particolare posizione rispetto alla superficie terrestre, si pensi ad esempio alle immagini fornite dai satelliti meteo, e in secondo luogo dalla possibilità di poter coprire ampie zone da monitorare. I satelliti per telerilevamento inoltre, percorrendo la propria orbita, ripassano sulla stessa area a intervalli di tempo regolari, consentendo in tal modo di avere un monitoraggio molto dettagliato dell'area di studio.
- Per completezza si citano infine i *satelliti* per l'osservazione dello spazio utili per le osservazioni astronomiche e i satelliti utilizzati per effettuare esperimenti tecnologici in assenza di gravità. Va detto inoltre che ognuno dei satelliti su elencati può essere ulteriormente classificato in *satellite militare* o *civile*. I satelliti militari sono utilizzati per scopi di difesa del territorio o per scopi di spionaggio (satelliti spia), essi in genere, per ovvi motivi, lavorano con segnali criptati e prestazioni migliori rispetto ai satelliti per uso civile. Una particolare menzione meritano i *satelliti GPS* utili per ottenere con estrema precisione la posizione di un qualsiasi punto sulla superficie terrestre. Essi sono utilizzati sia per scopi civili che per scopi militari. Per questi satelliti la trasmissione dei segnali per applicazioni civili può essere intenzionalmente degradata al fine di evitare di sfruttarne appieno le potenzialità.

Principali orbite dei satelliti

I satelliti nel loro moto intorno alla Terra (o a un qualsiasi altro corpo celeste) disegnano una traiettoria che viene definita orbita. Esistono diversi tipi di orbite e diversi parametri ad esse legati. Senza entrare in complessi dettagli che esulano dallo scopo di questo articolo è possibile suddividere in maniera semplificata le orbite dei satelliti in *orbite polari* e *orbite geostazionarie*. La figura 1 chiarisce il concetto di orbita polare: si tratta di un'orbita in cui il satellite nel suo moto sorvola a intervalli di tempo regolari il Polo Nord e il Polo Sud. Mentre il satellite percorre la propria traiettoria, la Terra continua a ruotare intorno al pro-

Figura 1 *Esempio di orbita polare*

Figura 2 *Esempio di orbita geostazionaria*

prio asse e di conseguenza il satellite ad ogni orbita osserverà aree differenti. I satelliti in orbita polare hanno la peculiarità di riuscire a coprire attraverso orbite successive l'intero globo terrestre ed è per questo che le orbite polari sono spesso utilizzate per l'osservazione della Terra. In tal caso i satelliti sono spesso posizionati ad altitudini molto basse (circa 700÷800 km) al fine di ottenere una risoluzione di osservazione ottimale. Per poter ricevere il segnale di un satellite in orbita polare occorre disporre di un'antenna che possa inseguire il satellite nel suo moto.

La figura 2 fa invece riferimento ad una tipica orbita geostazionaria: si tratta di un'orbita in cui il satellite sorvola costantemente l'equatore ad un'altitudine di circa 36000 km. A questa altitudine il satellite si muove con una velocità di rotazione uguale a quella della Terra, con un effetto complessivo che è quello in cui un osservatore posto sulla Terra "vede" il satellite geostazionario come se fosse fermo. I satelliti per le trasmissioni televisive sono posizionati su orbite geostazionarie e questo spiega perché le parabole montate sui tetti delle nostre case sono fisse e puntano sempre verso l'equatore.

Figura 3 *Denominazione delle bande dello spettro elettromagnetico*

IL TELERILEVAMENTO

Introduzione

L'osservazione della superficie terrestre ha assunto negli ultimi anni un'importanza sempre maggiore presso la comunità scientifica.

Di grande interesse in questo senso è l'attività di monitoraggio della superficie terrestre tramite sensori posti su piattaforme aeree o satellitari. Si parla in questi casi di telerilevamento, il cui scopo è fornire una descrizione completa e precisa di un certo insieme di grandezze fisiche e chimiche della zona in esame.

Uno dei principali vantaggi del telerilevamento è connesso con la possibilità di ottenere immagini da satellite ripetute nel tempo. Il satellite, infatti, percorrendo la propria orbita intorno al pianeta (non necessariamente la Terra), ritorna a intervalli di tempo costanti sulla stessa zona ed è quindi in grado di fornire immagini successive e regolarmente intervallate dell'area in corso di studio, permettendo così di seguire un fenomeno naturale (un fenomeno meteorologico, un'inondazione, un'eruzione vulcanica, una frana, ecc.) in tutta la sua evoluzione.

Le diverse tecniche di telerilevamento si distinguono sia in base alla porzione di spettro elettromagnetico esaminata, sia in base al tipo di sensore utilizzato.

Analizziamo separatamente questi due aspetti, anche se essi sono strettamente interdipendenti.

La scelta della porzione dello spettro elettromagnetico è di fondamentale importanza, poiché essa concorre alla determinazione dell'insieme delle grandezze fisiche e chimiche che si vogliono telerilevare. Utilizzando la suddivisione dello spettro elettromagnetico illustrata in figura 3, si può affermare, infatti, che la regione che comprende le microonde può essere presa in considerazione per l'analisi delle proprietà elettriche (costante dielettrica) e della rugosità della superficie del suolo, la regione nell'infrarosso compresa tra 10 e 15 µm è utile per ricavare la temperatura in superficie, mentre la regione del visibile e del vicino infrarosso è utilizzata per ricavare la composizione chimica del suolo.

Un altro importante fattore, che interviene nella scelta delle frequenze da adottare, è l'influenza degli effetti atmosferici sulla propagazione dell'onda elettromagnetica, i quali agi-

Figura 4 *Denominazione delle bande di frequenza alle microonde*

scono introducendo un'attenuazione. Tale attenuazione è legata all'interazione tra i segnali elettromagnetici utilizzati dai satelliti e le molecole che costituiscono l'atmosfera. Ad esempio con riferimento alle microonde, intorno ai 22 GHz si verifica una consistente attenuazione legata all'assorbimento delle molecole di vapor d'acqua, mentre in corrispondenza dei 60 GHz l'attenuazione raggiunge livelli molto alti a causa dell'assorbimento da parte delle molecole di ossigeno; di estrema importanza è infine la regione tra 1 e 10 GHz che presenta un'attenuazione quasi nulla ed è per questo che tale banda di frequenze è ampiamente utilizzata per il telerilevamento alle microonde.
Per quanto riguarda i tipi di sensore utilizzabili nelle tecniche di telerilevamento, è possibile classificarli essenzialmente in due grandi categorie: *sensori attivi* e *sensori passivi*. I sensori attivi (come i radar) trasmettono un segnale e ne misurano il segnale riflesso, i sensori passivi invece non trasmettono alcun segnale, ma sfruttano sorgenti di radiazioni esterne, come ad esempio il Sole, e si limitano a misurare il segnale riflesso.

Il telerilevamento radar

Il *telerilevamento radar* è una particolare tecnica di telerilevamento che si basa sulle tecnologie tipiche dei sistemi radar. Il radar lavora nella porzione di spettro elettromagnetico delle microonde e in particolare nelle bande L, S, C, X, K (vedere figura 4). Il motivo che ha portato alla nascita del telerilevamento radar è duplice: utilizzo della porzione di spettro alle microonde che consente di rilevare particolari caratteristiche del suolo e necessità di sopperire alle limitazioni intrinseche del telerilevamento ottico. Ci sono, infatti, diversi vantaggi nell'utilizzo delle microonde rispetto al visibile: le microonde hanno lunghezze d'onda diversi milioni di volte più grandi di quelle nel campo del visibile e quindi, a differenza della luce, riescono con facilità a penetrare la nebbia, le nuvole e altri ostacoli atmosferici.
Occorre inoltre aggiungere che la maggioranza dei sistemi ottici nel campo del visibile è capace di fornire immagini di un'area da studiare solo se essa è illuminata dal Sole e cioè di giorno. Questo perché i sistemi ottici sono in genere dei sistemi passivi, ossia non trasmettono alcun segnale, ma si limitano a ricevere il segnale - nel particolare caso la luce - trasmesso dal Sole. Il radar, invece, essendo un sensore attivo, si occupa oltre che della ricezione del segnale anche della sua trasmissione e questo garantisce la possibilità di disporre di immagini radar sia di giorno che di notte. La figura 5 è un esempio di immagine nel campo del visibile ripresa da un satellite. Dalla figura è evidente la presenza di nuvole che impedisce di osservare le aree sottostanti.

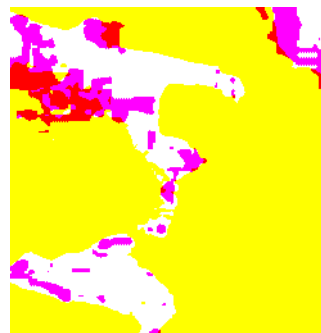



Figura 5 *Esempio di immagine ottica da satellite con presenza di nuvole*

IL RADAR AD APERTURA SINTETICA

Principio di funzionamento del radar

Prima di affrontare i concetti relativi ai radar ad apertura sintetica è doverosa una premessa sul funzionamento generale dei radar. RADAR è l'acronimo di RAdio Detection And Ranging (rivelazione e misura della distanza utilizzando onde radio). Tra le possibili configurazioni di sistemi radar consideriamone una tipica illustrata in figura 6. In questo schema il radar si compone di un'unica antenna puntata verso la direzione di osservazione; tale antenna come si vedrà si comporta sia da antenna trasmittente che ricevente (sensore attivo) e si parla in questi casi di radar monostatico. L'antenna invia un impulso a intervalli regolari di tempo e in corrispondenza di ogni impulso trasmesso si configura come antenna ricevente pronta a ricevere gli echi (segnali riflessi) generati da eventuali bersagli presenti lungo la direzione di osservazione. Quando si desidera esplorare tutte le possibili direzioni di osservazione l'antenna viene fatta ruotare su se stessa con una

velocità di diversi giri al minuto.
Il grafico in figura 7 illustra i segnali coinvolti in un sistema radar: il radar in un generico istante invia nel nostro caso un impulso sinusoidale di ampiezza **A** e dopo un periodo pari a **T** secondi giunge un eco generato dal bersaglio di figura 6. L'eco ha un'ampiezza inferiore ad **A** sia perché durante la propagazione il segnale trasmesso subisce un'attenuazione, sia perché quando il segnale giunge sul bersaglio quest'ultimo disperde una parte dell'energia del segnale trasmesso. In campo militare, per ovvi motivi, si cerca di progettare aerei che riescano a disperdere la maggiore quantità possibile di energia. Noto l'intervallo di tempo impiegato dal segnale trasmesso per raggiungere il bersaglio e ritornare all'antenna, indicato con **T** nella figura 7, si può calcolare la distanza del bersaglio dall'antenna secondo la formula: distanza = **c** × **T** / **2** dove **c** è la velocità di propagazione del segnale che può essere considerata pari alla velocità della luce (circa 300.000.000 m/s).

Figura 6 Schema di un tipico sistema radar

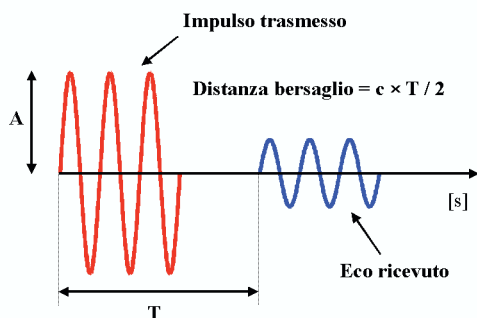



Figura 7 I segnali coinvolti in un sistema radar

Figura 8 Primo esempio di una situazione reale

La trattazione appena presentata fa riferimento ad una situazione del tutto ideale. Al fine di mettere in luce importanti limitazioni presenti in un sistema radar consideriamo una configurazione più vicina ai sistemi reali dove viene tenuto in conto il fascio di irradiazione dell'antenna. Questo significa che l'antenna pur puntando in una ben determinata direzione trasmetterà anche in altre direzioni secondo il proprio fascio di irradiazione. Analogamente l'antenna non riceverà i soli echi provenienti dalla direzione di puntamento, ma da tutte le posizioni coperte dal fascio di irradiazione. Il bersaglio **B2** di figura 8 ad esempio pur non essendo lungo la direzione di osservazione genera un eco che verrà ricevuto dal radar, il quale a sua volta lo "confonderà" con un bersaglio presente lungo la direzione di osservazione.
Il bersaglio **B3** invece pur se posizionato lungo la direzione di puntamento è troppo distante dall'antenna per cui è raggiunto da impulsi troppo attenuati per generare un eco percettibile in ricezione, si dice in tal caso che il bersaglio è fuori dalla portata del radar. Infine il bersaglio **B4** anche se è all'interno del fascio dell'antenna, non sara rivelato dai radar poiché l'eco generato da esso viene riflesso verso una direzione che non è quella dell'antenna ricevente. Da questa situazione ne deriva il diagramma temporale illustrato in figura 9 dove solo gli echi di **B1** e **B2** sono "visibili". Si noti come l'eco di **B2** giunga prima poiché tale bersaglio è più vicino. Si osservi inoltre che gli echi non giungono con uguale ampiezza. Esaminiamo un altro caso interessante illustrato in figura 10. Si tratta di due bersagli posizionati alla stessa distanza dall'antenna. In questa situazione i due echi giungeranno contempo-

raneamente all'antenna ricevente mescolandosi fra loro in un unico eco. Il radar non sarà quindi in grado di distinguere i due echi e considererà come un unico bersaglio i bersagli **B1** e **B2**. In questi casi si dice che la *risoluzione in azimuth* del radar è tropppo bassa per distinguere i due bersagli. Questa limitazione viene spesso sfruttata nelle strategie militari di attacco quando più aerei viaggiano vicini fra loro al fine di farsi riconoscere dal radar come un unico aereo e indurre un allarme sottostimato nei sistemi di difesa.

Principio di funzionamento del radar ad apertura sintetica

Nel paragrafo precedente si è osservato che a causa della limitata risoluzione in azimuth il radar non riesce a distinguere due bersagli che si trovano alla stessa distanza da esso ma in posizioni differenti.
Per migliorare la risoluzione in azimuth del radar occorre restringere il fascio di irradiazione dell'antenna, e ciò può essere ottenuto o utilizzando un'antenna più grande o anche costruendo una schiera di antenne ossia collegando opportunamente più antenne fra loro. Si veda a tal proposito la figura 12.

Vediamo ora come i concetti base su esposti si adattino ai radar montati su satellite per scopi di telerilevamento. A tal proposito si faccia riferimento alla figura 13: il satellite viene posizionato in genere ad una altitudine di circa 700÷800 km e l'antenna utilizzata è in genere un'antenna planare di dimensioni che nel caso del satellite europeo ERS-1 sono 10m in larghezza e 1m in altezza. Essa viene fissata saldamente al satellite e quindi non può ruotare, tuttavia in alcuni sistemi è possibile direzionare elettronicamente il fascio verso la direzione desiderata alimentando con fasi opportune i vari elementi che compongono l'antenna. Grazie al moto del satellite lungo la propria orbita, l'antenna, pur restando fissa, riesce a osservare aree a terra differenti. I bersagli in questo caso saranno composti da case, strade, pali, fiumi, ecc., ognuno dei quali genererà un proprio eco.
Il problema che si pone anche in questo sistema è riuscire a distinguere due bersagli vicini fra loro. Nel caso di due edifici vicini fra loro accadrà che gli echi da essi generati giungeranno al radar nello stesso istante e di conseguenza il radar li riconoscerà come un singolo edificio. Occorre inoltre considerare che l'antenna si trova ad una notevole distanza dall'area da osservare (circa 700÷800 km), e poiché il fascio dell'antenna si allarga allontanandosi da essa vuol dire che al suolo il fascio sarà notevolmente esteso con conseguente scarsa risoluzione in azimuth.
Per migliorare la risoluzione non si può pensare di utilizzare un'antenna più grande per ovvie ragioni di ingombro e di peso. Non è possibile nemme-

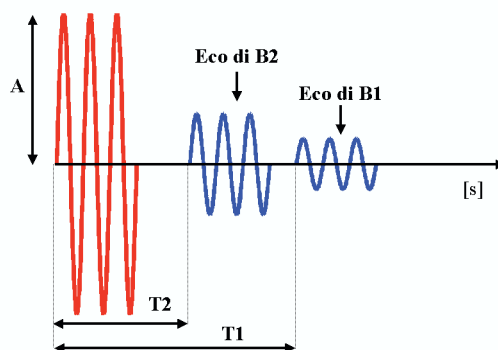



Figura 9 *Diagramma temporale relativo alla figura 8*

Figura 10 *Secondo esempio di una situazione reale*

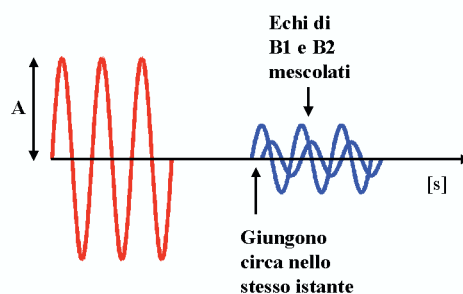



Figura 11 *Diagramma temporale relativo alla figura 10*

no adottare la soluzione della schiera di antenne ossia tentare di utilizzare più satelliti ognuno con una propria antenna e di combinare insieme i segnali ricevuti da essi, sia perché ne verrebbe fuori un sistema estremamente costoso, sia perché non è possibile posizionare i satelliti in orbita con la precisione desiderata evitando, fra l'altro, il rischio di collisione fra essi.

La soluzione economica ed efficace che si adotta è sfruttare il movimento del satellite lungo la sua orbita: mentre il satellite è in movimento l'antenna occupa posizioni differenti in istanti differenti. Si può dimostrare che l'antenna occupando posizioni differenti simula la presenza di più antenne, ossia simula il comportamento di una schiera di antenne con conseguente possibilità di migliorare la risoluzione in azimuth.

Si dice in tal caso che si è effettuata la sintesi di un'apertura dell'antenna più grande (fascio più stretto) rispetto a quella reale, da cui il nome di radar ad apertura sintetica o SAR (Synthetic Aperture Radar). L'elaborazione degli echi ricevuti, che viene effettuata per simulare la schiera di antenne con un'unica antenna, è molto complessa ed è tuttora oggetto di studio presso la comunità scientifica.

ELABORAZIONE DI IMMAGINI SAR

Le immagini radar

Le immagini radar ottenute dopo una elaborazione che simula la schiera di antenne, secondo il principio su esposto, si compongono di una immagine di *ampiezza* e una immagine di *fase*. In questo paragrafo ci occuperemo dell'immagine di ampiezza: essa è un'immagine composta da diversi punti detti pixel. Ogni pixel fa riferimento a un ben determinato punto (bersaglio) a terra. Il pixel viene rappresentato in scala di grigi in modo tale che quanto più esso appare bianco tanto più è intenso l'eco che esso genera verso l'antenna radar. Al contrario un pixel nero indica un bersaglio al suolo che ha assorbito il segnale trasmesso o lo ha riflesso verso una direzione che non è quella dell'antenna radar. L'immagine di fase é anch'essa un'immagine composta da diversi punti detti pixel. Ogni pixel fa riferimento a un ben determinato punto a terra, e rappresenta un valore che è la fase con cui l'eco arriva al radar. Questo valore di fase viene utilizzato per calcolare la distanza del bersaglio dal satellite. La figura 14 illustra in maniera semplificata alcuni casi di bersagli al suolo e in corrispondenza di ognuno di essi viene riportato il colore che esso assumerebbe in una immagine di ampiezza: i pixel bianchi rappresentano un forte eco verso il radar, mentre i pixel neri rappresentano un eco debole o inesistente verso il radar.

Si noti come una montagna appaia con colori vicini al bianco dalla parte del pendio rivolto verso il satellite e invece generi dei pixel scuri dalla parte del pendio opposto. In quest'ultimo caso, o l'eco generato dal pendio viene riflesso verso una direzione che non coincide con quella del satellite o gli impulsi trasmessi dal radar non riescono a raggiungere tale area
(pendio in ombra) perchè bloccati dal pendio rivolto verso il satellite. I grandi edifici in generale appaiono molto luminosi poiché la loro strut-

Figura 12 Confronto tra fasci di irradiazione

Figura 13 Schema di acquisizione di immagini radar da satellite

Figura 14 *Risposta radar di alcuni tipi di bersaglio*

Figura 15 *Ampiezza di una immagine radar acquisita dal satellite ERS-1*

tura è tale da generare quasi sempre un forte eco diretto verso il satellite.
Le strade così come il mare calmo si comportano da specchio e in genere appaiono scure perché l'impulso trasmesso dal radar viene riflesso da tali bersagli verso altre direzioni. Una nave (che in genere produce un eco anche verso il radar) in mezzo ad un mare calmo può essere quindi facilmente riconosciuta come uno o più pixel bianchi circondati da una moltitudine di pixel più scuri. La figura 15 è un esempio reale di ampiezza di una immagine acquisita dal radar ad apertura sintetica europeo ERS-1.
L'immagine fa riferimento ad un'area vicino la città di Sulmona. Si noti come il capannone industriale sia perfettamente riconoscibile grazie al forte eco da esso generato. Si noti inoltre come le colline generino il classico alternarsi di zone chiare (pendio rivolto verso il SAR) e zone scure (pendio opposto alla direzione del SAR). Nell'immagine sono infine facilmente riconoscibili numerose strade, le quali per quando affermato in precedenza appaiono in genere come delle linee scure.

L'interferogramma

In questa sezione cercheremo di capire cosa succede se un radar ad apertura sintetica osserva la stessa area da posizioni differenti. È noto che l'essere umano riesce ad avere un percezione tridimensionale di quello che osserva perché possiede due occhi che gli permettono di osservare lo stesso oggetto da due posizioni leggermente differenti. Se l'uomo possedesse un solo occhio vedrebbe un paesaggio così come appare in fotografia, cioè senza la profondità. Anche nel caso del radar accade qualcosa di simile: si è visto infatti che l'immagine radar è una immagine bidimensionale. Per ottenere una immagine tridimensionale della zona osservata occorre, come nel caso dell'essere umano, osservare la stessa area da due posizioni (angoli di vista) differenti. È possibile fare ciò sia simultaneamente, usando due radar montati sulla medesima piattaforma, sia in differenti istanti utilizzando, in questo caso, i passaggi ripetuti dello stesso SAR sulla zona considerata.
Avendo acquisito due immagini radar della stessa zona con due angoli di vista leggermente differenti, è possibile effettuare un confronto tra la distanza di uno stesso bersaglio dalla prima e dalla seconda antenna. Da questo confronto tramite semplici considerazioni geometriche è possibile stabilire con precisione la quota a cui si trova il bersaglio (un edificio, una strada, ecc.). Effettuato questo calcolo per ogni punto dell'immagine si riesce ad ottenere una mappa completa delle quote dell'area osservata dal satellite nota come *mappa topografica*. Si è detto nel precedente paragrafo che la distanza del bersaglio dal radar è codificata nella fase del pixel che lo rappresenta. Si intuisce quindi che per ottenere una mappa topografica del terreno occorre utilizzare l'immagine di fase e in particolare due immagini di fase osservate con angoli di vista differenti per ottenere la visione tridimensionale. Quello che si fa in questi casi è fare una sottrazione tra la prima immagine di fase e la seconda, il risultato è una

Figura 16 *Esempio di area osservata da due SAR con angoli di vista differenti*

Figura 17 *Interferogramma relativo all'area di figura 16*

terza immagine di fase nota come *interferogramma*. Supponiamo ad esempio di osservare con due SAR con angoli di vista differenti l'area illustrata in figura 16. Sottraendo le due immagini di fase ottenute si ottiene l'interferogramma di figura 17. È importante osservare come l'interferogramma ottenuto sia strettamente legato alle quote del terreno, in particolare le curve (dette frange) visibili sono riconducibili a curve di livello, ossia ogni curva di un ben determinato colore rappresenta una ben determinata quota.
Se, quindi, a partire dall'interferogramma, si desidera ottenere la mappa di quota dell'area osservata occorre trasformare queste curve di livello in una mappa topografica.
Tale processo si chiama Phase Unwrapping ossia srotolamento della fase e consiste nel trasformare (srotolare) i valori di fase dell'interferogramma (che sono compresi tra 0° e 360°) in valori di quota.

lizzando tecniche interferometriche. In particolare per ottenere la visione tridimensionale con il radar sono state utilizzate due antenne: la prima collocata all'interno dello Shuttle (operante sia in trasmissione che in ricezione) e la seconda collocata fuori dallo Shuttle e agganciata ad esso mediante un braccio meccanico lungo circa 60 metri (figura 17b). Quest'ultima antenna lavora solo in ricezione ed in particolare riceve gli echi trasmessi dalla prima antenna. La configurazione globale prevede quindi una sola antenna trasmittente e due antenne riceventi per ottenere la visione tridimensionale.
Le mappe topografiche relative a questa missione sono disponibili gratutitamente (anche se degradate nella risoluzione) al sito web: http://www2.jpl.nasa.gov/srtm/

L'attualità della tecnica su esposta è testimoniata da una missione recentemente condotta dall'agenzia spaziale americana (NASA) in collaborazione con altre agenzie spaziali, compresa l'agenzia spaziale italiana (ASI).
La missione è denominata SRTM (Shuttle Radar Topography Mission) e tra i suoi obiettivi principali compare quello di produrre una mappa topografica (mappa di quote) di quasi tutto il globo terrestre uti-

Figura 17b *Lo shuttle utilizzato per produrre mappe topografiche*

Misura di deformazioni al suolo

Le potenzialità del radar ad apertura sintetica non sono semplicemente confinate alla generazione di mappe di topografia del terreno, ma è possibile estrarre altre interessanti informazioni sulla area illuminata dal radar.
Si supponga infatti di disporre di quattro immagini radar delle quali due acquisite in un certo anno e altre due acquisi-

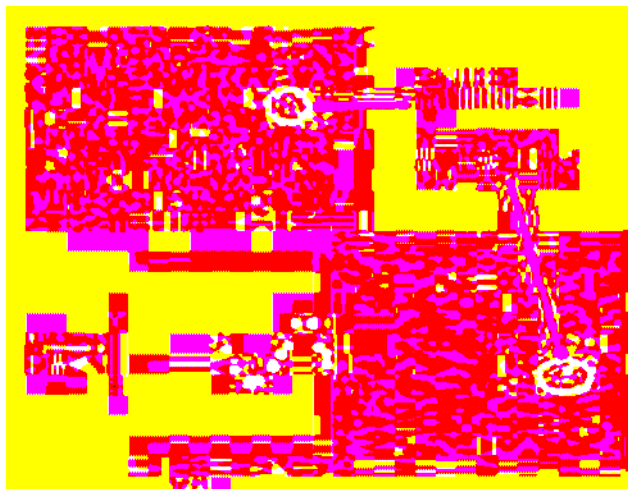



Figura 18 *Esempio di misura di deformazioni millimetriche*

te in un altro anno. Le prime due immagini si possono utilizzare ad esempio per ottenere una prima mappa topografica del terreno come è stato spiegato nel paragrafo precedente. Le altre due immagini potrebbero anch'esse essere utilizzate per generare un'altra mappa topografica. Cosa accade a questo punto se le due mappe ottenute sono differenti? La risposta è che se le due mappe di quota sono differenti la spiegazione è da cercare in eventuali deformazioni del terreno intervenute nel periodo di tempo che separa le due coppie di immagini.
Tramite questa semplice considerazione si capisce che disponendo di numerose immagini (diverse decine) acquisite in periodi differenti è possibile capire se nell'area osservata si sono verificate delle deformazioni ed eventualmente valutarne l'entità.
L'utilizzo di questa tecnica trova spazio ad esempio nello studio delle deformazioni subite dal terreno a causa di un terremoto: in tal caso si genera una mappa iniziale di quote prima dell'evento sismico ed una seconda mappa di quote ottenuta con due immagini acquisite in date successive a quella del terremoto. Oltre a deformazioni violente è possibile anche studiare deformazioni molto più graduali: molti edifici ad esempio rischiano di crollare a causa della instabilità del terreno sottostante. In questi casi prima del crollo vi può essere una fase che può durare anche anni in cui il terreno lentamente sprofonda verso il basso con velocità pari a qualche mm per anno. La tecnica presentata è capace di stimare anche queste piccole variazioni di quota per cui, disponendo di numerose acquisizioni radar nel corso degli anni, è possibile capire se vi sono edifici su terreni instabili che lentamente stanno sprofondando e di conseguenza è possibile allertare con sufficiente anticipo la protezione civile.
La figura 18 mostra un caso reale: si tratta di un capannone in metallo del quale è illustrata in alto una foto aerea ad alta risoluzione e in basso a destra la corrispondente immagine radar di ampiezza. In basso a sinistra si mostrano le oscillazioni millimetriche subite dal capannone durante il corso degli anni. Non è ancora ben chiaro in questo specifico caso se le deformazioni siano dovute al terreno sottostante o alla dilatazione/compressione della struttura metallica del capannone al variare della temperatura dell'ambiente.
Si noti come un satellite posizionato a centinaia di km di altezza sia capace di misurare deformazioni millimetriche!

CONCLUSIONI

In questo articolo sono stati illustrati in breve i concetti di base di funzionamento del SAR e le potenzialità offerte da questo strumento di misura nella prevenzione di gravi rischi ambientali. Molti degli argomenti presentati sono ancora oggetto di studio presso molte università italiane e internazionali e presso le agenzie spaziali. Il lettore interessato ad approfondire queste tematiche può far riferimento alla seguente bibliografia:

- Libro: Appunti di teoria e tecniche radar, A. Monti Guarnieri, L. Zanzi, Ed. Cusl
- Sito (in italiano) ASI (Agenzia Spaziale Italiana): http://www.asi.it/
- Sito (in inglese) ESA (Agenzia Spaziale Europea): http://earth.esa.int/
- Sito (in inglese) NASA (Agenzia Spaziale Americana): http://www.jpl.nasa.gov/

Intelligenza artificiale

Le Reti Neurali sono una delle più interessanti creature nate dalla ricerca sull'Intelligenza Artificiale. In questo articolo vedremo in dettaglio cosa sono, come funzionano, e quali sono le loro principali applicazioni. Nel prossimo invece daremo un'occhiata più approfondita agli aspetti implementativi ed a quelli pratici legati al loro impiego.

Partiamo da una constatazione: esistono delle operazioni che i normali calcolatori (di qualsiasi grandezza e potenza di calcolo) non riescono ad eseguire in maniera efficiente. Basti pensare al riconoscimento di immagini o di suoni, alla classificazione automatica di dati, all'apprendimento, alla generalizzazione dei dati appresi, etc. Al contrario il cervello umano (o in genere animale), sebbene fisiologicamente molto più lento, riesce in questi compiti molto meglio e spesso anche più velocemente. Perché allora non cercare in qualche modo di imitarlo? Fu proprio questa l'idea che più di 50 anni fa portò McCulloch e Pitts (ed in seguito tanti altri) alla creazione ed allo studio delle reti neurali.
Una rete neurale consiste in un modello matematico che cerca di imitare la struttura ed alcuni meccanismi tipici del cervello, e grazie a questa somiglianza riesce a riprodurne alcune interessanti caratteristiche, come la capacità di apprendere, di riconoscere e di generalizzare.
A differenza di altri sistemi di calcolo, una rete neurale non viene programmata per eseguire un algoritmo, bensì è addestrata a svolgere un determinato compito, semplicemente mostrandole degli esempi sotto forma di coppie ingresso-uscita. La rete impara così ad associare ad ogni ingresso un'uscita, e anche in una certa misura a generalizzare quello che ha imparato. E' sufficiente quindi che gli ingressi "somiglino" o abbiano qualche relazione con quelli usati durante l'addestramento per essere riconosciuti o valutati correttamente. Grazie a questa capacità le reti neurali sono state utilizzate con successo in applicazioni quali il riconoscimento dei caratteri scritti a mano, e delle immagini, la sintesi ed il riconoscimento vocale, l'analisi automatica di dati (chimici, biologici…), la traduzione automatica, il controllo di robot, la guida automatica di autoveicoli, e tanto altro…
Come vedremo implementare una rete neurale non è eccessivamente complicato, per cui la conoscenza di questo potente strumento apre moltissime possibilità anche in campo hobbistico.

Com'è fatta una Rete Neurale

In analogia al cervello, una rete neurale è formata da un certo numero di semplici elementi di calcolo (**i neuroni**), fortemente interconnessi tra loro. Le interconnessioni (che imitano strutture naturali quali gli **assoni**, i **dendriti** e le **sinapsi**) come vedremo svolgono un ruolo fondamentale nel processo di apprendimento. Le capacità di una rete dipendono sia dal numero di neuroni utilizzati, sia soprattutto dalla sua topologia, cioè da come sono connessi tra loro i neuroni. Esistono infinite topologie possibili, ma si può fare una prima grossa suddivisione (figura 1) tra reti alimentate in avanti (feed for-

di Antonio Di Stefano
a.distefano@farelettronica.com

Introduzione alle reti neurali

Figura 1 Esempio di reti neurali: a) feed forward, b) ricorrente

ward) e reti ricorrenti (o ricorsive).
Nelle prime non esiste nessun collegamento che riporti in qualche modo le uscite in ingresso, quindi i segnali procedono soltanto dagli ingressi alle uscite. Questo le rende abbastanza semplici da studiare ed addestrare, ma non gli permette di mantenere uno "stato" interno, cioè una memoria di quello che è successo nel passato. Questo tipo di rete è il più studiato ed utilizzato. Le reti ricorrenti invece potrebbero avere delle capacità più estese, però non sono state ancora ben comprese e non si conoscono sempre dei metodi adatti per addestrarle, per cui sono molto meno utilizzate. Esistono inoltre delle reti un po' particolari, che è difficile inquadrare in una di queste due categorie, che saranno descritte brevemente più avanti. Tutte le reti neurali presentano alcune caratteristiche comuni che le rendono ulteriormente interessanti da un punto di vista applicativo: esse sono tolleranti agli errori, cioè riescono a fornire l'uscita corretta anche in presenza di dati parzialmente errati o corrotti; sono intrinsecamente fault tolerant, cioè continuano a funzionare anche se si verificano dei guasti o degli errori interni; in presenza di errori o guasti le prestazioni della rete peggiorano gradualmente (graceful degradation), non si ha una interruzione brusca del funzionamento come nei sistemi di calcolo tradizionali.
Per limitare l'ambito della nostra discussione ci concentreremo di seguito soltanto sui tipi di reti più diffusi e più facilmente utilizzabili.

Figura 2 Percettrone semplice

IL PERCETTRONE

Consideriamo la rete più semplice che è possibile immaginare: quella formata da un solo neurone. Questa rete viene chiamata comunemente "percettrone", e può essere considerata come il mattoncino elementare con cui costruire reti più complesse. Il neurone, come accennato nel paragrafo precedente è un semplice elemento di calcolo, dotato di un certo numero di ingressi e di un uscita (figura 2).
Il comportamento del neurone è il seguente: esso si attiva (cioè dà in uscita un livello "alto") solo quando i segnali ricevuti dagli ingressi superano una certa soglia di attivazione, altrimenti l'uscita rimane inattiva (a livello "basso"). I segnali d'ingresso però non hanno tutti la stessa importanza, infatti ad ognuno di essi è attribuito un "peso", che fa si che il suo contributo valga più o meno, e che sia positivo o negativo. Da un punto di vista matematico quindi l'ingresso complessivo al neurone può essere calcolato con la seguente formula:

$$I = i_0w_0 + i_1w_1 + \ldots i_nw_n$$

dove gli i_x sono i valori assunti dai vari ingressi, ed i w_x i pesi associati a ciascuno di essi (per

Figura 3 Funzioni di attivazione a gradino, lineare e sigmoide

generalità possiamo supporre che essi siano numeri reali, ma questo non è strettamente necessario).

Calcolato l'ingresso complessivo (I), occorre decidere il valore che assume l'uscita (questo valore si chiama **attivazione** del neurone).

Per fare questo si utilizza una funzione matematica, detta appunto **funzione di attivazione**, che in genere ha una forma a "gradino".

Le funzioni più comunemente utilizzate per questo scopo sono la funzione **gradino** vera e propria, la funzione **lineare a tratti**, e la funzione **sigmoide** (in figura 3 sono rappresentate le funzioni e le loro espressioni matematiche). In genere si utilizzata la prima quando si ha bisogno di un uscita di tipo "digitale", mentre la seconda e la terza quando si vogliono delle uscite a valori continui (uscita di tipo "analogica"). Come si può vedere nel caso della funzione lineare, l'attivazione può assumere anche valori negativi se necessario (anche le altre due possono essere rese simmetriche rispetto all'origine).

Riassumendo, per calcolare l'uscita del neurone è sufficiente prima calcolare la somma pesata degli ingressi, e poi con questa calcolare il valore assunto dalla funzione di attivazione g(I). Semplice no?

Ma come fa il neurone ad imparare? La risposta è proprio nei pesi wx. Scegliendo opportunamente i valori di questi coefficienti è possibile trasmettere alla rete (o farle acquisire) delle conoscenze. Per vedere come questo sia possibile consideriamo un semplice esempio. Supponiamo di utilizzare un percettrone dotato di due ingressi e di un uscita, e di volergli insegnare a comportarsi come una porta AND. Supponiamo che gli ingressi siano di tipo digitale (cioè i cui possibili valori siano soltanto 0 e 1, o comunque limitati in questo intervallo), e supponiamo di utilizzare una funzione di attivazione a gradino, con soglia di 1,5 (cioè se $I < 1,5$ allora g=0, se $I \geq 1,5$ allora g=1), come mostrato in figura 4.

Per ottenere il risultato voluto basta attribuire ai pesi il valore 1. Infatti si ha:

$I = 1*i1 + 1*i2$

che chiaramente è maggiore di 1,5 (uscita attiva) solo quando entrambi gli ingressi valgono 1. Se avessimo voluto insegnare alla rete la funzione OR invece della AND, sarebbe stato sufficiente attribuire ai pesi il valore 2 (o comunque un numero maggiore di 1,5). Per ottenere la funzione NOT avremmo dovuto spostare la soglia della funzione di attivazione a –0.5 ed usare un peso pari a -1, oppure, come si fa comunemente, usare un **ingresso**

Figura 4 Implementazione delle funzioni logiche tramite percettroni

di ***bias*** (di polarizzazione), cioè un ingresso "fittizio" permanentemente collegato ad 1, che è in grado di dare un contributo fisso, ma programmabile grazie al suo peso. Questo accorgimento è molto utilizzato, in quanto evita di dovere modificare la funzione di attivazione, e permette invece di regolare la soglia agendo soltanto su un peso. Va sottolineato comunque che normalmente i valori da assegnare ai pesi non sono scelti manualmente, ma si utilizzano per questo degli algoritmi di addestramento (che vedremo più avanti) per farli apprendere alla rete in base a degli esempi.

I percettroni possono imparare anche altre funzioni oltre a quelle booleane, in particolare se si utilizza una funzione di attivazione lineare il neurone è in grado di funzionare da filtro digitale! E quindi ad esempio è in grado di fare delle interpolazioni o delle previsioni sui dati, di rimuovere il rumore, di eseguire equalizzazioni automatiche, etc. Esistono però anche delle funzioni che non sono rappresentabili da un singolo neurone (funzioni "non linearmente separabili"), per esempio la funzione XOR. Per superare questo limite occorre utilizzare reti più complesse.

RETI DI PERCETTRONI

Complichiamo un po' la nostra rete come mostrato in figura 5: questa volta sono stati utilizzati più percettroni semplici, i cui ingressi sono collegati in comune, e le cui uscite sono indipendenti. Il numero di ingressi e di uscite (e quindi neuroni) è arbitrario e dipende dall'applicazione. Tutto funziona quasi come nel caso precedente, infatti in pratica ogni neurone è indipendente dagli altri (tranne che per gli ingressi), e quindi ogni uscita può essere calcolata separatamente.
Nonostante la semplicità di questa rete, essa è una delle più utilizzate, e le sue applicazioni sono veramente innumerevoli! In particolare questo tipo di rete viene utilizzata per riconoscere e classificare i dati forniti in ingresso. Ad esempio è possibile utilizzarla per il riconoscimento ottico dei caratteri. In questo caso gli ingressi ricevono i dati relativi ai "pixel" dell'immagine che contiene il carattere, e le uscite, che saranno tante quanti sono i caratteri da riconoscere, indicheranno con la loro attivazione qual è il carattere riconosciuto (figura 6). Se si utilizza una funzione di attivazione più graduale (lineare o sigmoide ad esempio), si può avere anche una stima della "certezza" con cui è stato riconosciuto il carattere. La rete viene addestrata presentandole i vari caratteri (anche in versione un po' deformata) ed indicandole di volta in volta la risposta corretta. Questo tipo di rete viene utilizzata anche in strumenti quali spettrometri, matrici di sensori e nasi elettronici per riconoscere i pattern a cui sono associate determinate specie chimiche. Il funzionamento è lo stesso: i dati rilevati vengono presentati in ingresso, e ad ogni uscita si associa una particolare risposta (cioè una particolare specie chimica o odore.
Spesso viene inclusa anche l'uscita relativa alla condizione di "non riconosciuto").
Nel campo della robotica la rete può essere utilizzata per associare ai dati provenienti da sensori di vario tipo la migliore azione da compiere (in termini di controllo dei movimenti per esempio), per implementare sistemi di controllo non lineare, o per far comprendere al robot la situazione in cui esso si trova a partire dai dati raccolti. La capacità di generalizzazione della rete, permette di ottenere in questi casi comportamenti meno "rigidi", spesso soddisfacenti anche in presenza di situazioni impreviste.

ADDESTRAMENTO DEI PERCETTRONI

Per fare in modo che la rete impari a partire dagli esempi si utilizzano particolari algoritmi di addestramento. Lo scopo di questi algoritmi è quello trovare un insieme di pesi tali che la rete commetta il minor numero di errori possibile (cioè riconosca bene gli esempi con cui è stata addestrata). In genere il funzionamento è il seguente: si presenta alla rete un esempio tra quelli scelti per l'addestramento (***training set***), si calcola la sua uscita, si calcola l'errore, cioè quanto l'uscita è differente da quella attesa (***terget***), e si correggono i pesi in modo da far diminuire l'errore.

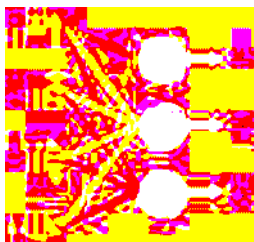



Figura 5 Esempio di rete di percettroni

Consideriamo inizialmente un percettrone singolo. Inizialmente i pesi della rete saranno scelti casualmente (per esempio nell'intervallo tra –0,5 ed 0,5), si presenta quindi un esempio agli ingressi e si calcolata l'uscita della rete e l'errore rispetto al target. In formule:

- Calcolo ingresso complessivo: $I = i_0w_0 + i_1w_1 + \ldots i_nw_n$
- Calcolo uscita: O=g(I)
- Calcolo errore: Err=T-O

T è il valore atteso (target) ed O l'uscita del percettrone (la sua attivazione). La ragola per aggiornare i pesi è la seguente (valida per le funzioni di attivazione viste):

$W_j=W_j + \alpha*ij*Err$

Dove α è una costante detta **learning rate**, che regola la velocità di apprendimento (scelta arbitrariamente, di solito dell'ordine di 0,1), ed i_j è il j-esimo ingresso (quello a cui si riferisce il peso in questione).

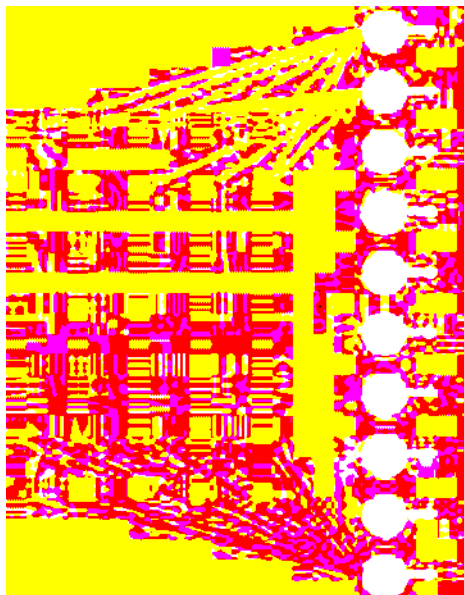



Figura 6 Rete per il riconoscimento ottico dei caratteri numerici

L'operazione va ripetuta un numero di volte prestabilito (presentando esempi diversi), oppure fino a quando l'errore non scende sotto una soglia prefissata.
Per le reti di percettroni l'algoritmo è identico, solo che bisogna applicarlo a ciascun neurone.

PERCETTRONI MULTISTRATO

Consideriamo adesso la rete rappresentata in figura 7. Essa è simile ad una rete di percettroni, ma in questo caso sono presenti più "strati" di neuroni, per questo motivo questo tipo di reti vengono chiamate percettroni multistrato (*Multi-Layer Perceptrons*, in sigla MPL).
Il numero di strati, e di neuroni per ogni strato è arbitrario, così come la possibilità di eseguire tutti i collegamenti o solo parte di essi.
I neuroni degli strati intermedi vengono detti "nascosti", poiché non hanno connessioni con l'esterno.

Grazie alla loro maggiore complessità, queste reti neurali presentano maggiori capacità rispetto alle precedenti. In particolare oltre ad una maggiore abilità nell'imparare relazioni complesse tra gli ingressi e le uscite, esse sono in genere in grado di elaborare in modo più sofisticato le informazioni, di individuare similitudini e caratteristiche comuni, di fare previsioni, ed anche di generalizzare ad un livello un po' più alto ciò che hanno appreso. Grazie a queste caratteristiche, gli MLP sono in assoluto le reti neurali più utilizzate, ed hanno trovato impiego nei campi e nelle applicazioni più varie.
Fare funzionare "in avanti" (cioè ottenere le uscite a partire dagli ingressi) gli MLP è relativamente semplice, infatti si procede come nei casi precedenti, basta solo usare come input per i vari strati le uscite di quelli precedenti. Ben più problematica è la fase di addestramento. Il problema è che si può calcolare l'errore solo per l'ultimo strato (e quindi aggiustare i suoi pesi di conseguenza), ma non per gli strati intermedi. Questo ostacolo ha fermato la ricerca nel campo delle reti neurali per almeno 20 anni! Solo negli anni '80 è stato

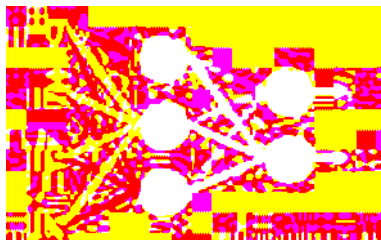



Figura 7 Esempio di Percettrone multistrato con uno strato nascosto

proposto un algoritmo chiamato **back propagation** che riesce a risolvere il problema. Maggiori dettagli su questo algoritmo verranno dati nella prossima puntata di questa serie. Esistono anche altre tecniche per addestrare gli MPL, in particolare meritano un cenno quelle che utilizzano degli algoritmi di ottimizzazione.
Ad esempio è possibile utilizzare un algoritmo genetico per selezionare il migliore set di pesi per minimizzare l'errore della rete per un certo training set. Lo svantaggio di questi algoritmi è che in genere sono abbastanza più lenti della back propagation.
Come già detto le applicazioni degli MLP sono pressoché infinite, ad esempio essi sono molto utilizzati in tutti quei casi in cui occorra associare a degli ingressi alcune uscite dipendenti da essi in modo molto complesso o poco chiaro analiticamente. Pensiamo ad esempio al controllo di alcuni processi industriali complessi, o all'attribuzione di una misura di qualità ad alcuni particolari set di dati (nel campo della spettrografia, in quello alimentare, in quello biomedico, etc.).
In questi casi è molto più semplice insegnare alla rete a fornire le stesse risposte che avrebbe dato un esperto umano che non sviluppare un complicato (ed in genere poco soddisfacente) modello matematico. Un esempio abbastanza rappresentativo di questo approccio è ad esempio l'utilizzo di un MLP per analizzare costantemente i tracciati elettrocardiografici (ECG) e capire in tempo se si stanno per manifestare episodi di aritmia o anche patologie più gravi. In questo caso (e negli altri in cui è necessario analizzare l'evoluzione di un segnale nel tempo), i campioni del segnale vengono fatti "scorrere" sugli ingressi, come mostrato in figura 8. Per chi fosse interessato a questo tipo di applicazione, sono disponibili su Internet dei set di dati da utilizzare per l'addestramento (ad esempio quelli del MIT-BIH Arrhythmia Database).

Figura 8 Elaborazione di serie temporali (segnali funzione del tempo)

Utilizzando gli MPL è possibile affrontare anche problemi legati al controllo del movimento di bracci robotici complessi (che richiederebbero pesanti calcoli legati alla cinematica inversa), o alla guida di autoveicoli (ad esempio il parcheggio di un auto in retromarcia).
L'addestramento avviene utilizzando come esempi la posizione attuale e da quella voluta (ingressi), e come target (uscita) il movimento da compiere in quella situazione. In genere è sufficiente utilizzare come esempio un insieme abbastanza limitato di posizioni, la rete sarà in grado di comportarsi bene anche in quelle intermedie non mostrate ed anche in alcune non previste completamente dall'addestramento.
Un'altra sorprendente applicazione in questo campo è "ALVINN", una rete sviluppata alla Carnegie Mellon University, che riesce a guidare autonomamente un furgoncino su un'autostrada. Per fare questo la rete riceve un'immagine della strada riscalata a 32x32 pixel, che viene fornita gli ingressi di un MLP.
La rete ha uno strato nascosto non totalmente connesso, ed un certo numero di uscite, ciascuna delle quali rappresenta una posizione dello sterzo.
Dopo avere osservato per un po' il comportamento di un guidatore umano, il sistema impara ad associare alle immagini della strada

Figura 9 Esempio di reti neurali di a) Hopfield, b) Kohonen

la corretta posizione dello sterzo. In letteratura è possibile trovare moltissime altre applicazioni che non è possibile citare qui per ragioni di spazio.

ALTRI TIPI DI RETI

Esistono altri tipi di reti neurali che differiscono molto, strutturalmente e funzionalmente, da quelle viste finora. Tra queste vale la pena di citare le reti di Hopfield e quelle di Kohonen (figura 9).

Le prime sono delle reti costituite da un solo strato di neuroni, in cui l'uscita di ciascun neurone è collegata agli ingressi di tutti gli altri (si tratta quindi di una rete ricorrente).
Le reti di Hopfield risultano particolarmente adatte ad implementare delle memorie autoassociative, che cioè riescono a fornire in uscita le esatte configurazioni imparate in fase di addestramento quando agli ingressi ne viene presentata una versione distorta o corrotta da rumore.
La rete viene utilizzata ad esempio per riconoscere immagini degradate in bianco e nero (ad un solo bit colore) e restituire l'originale. Un risultato teorico interessante è che la rete può memorizzare in maniera affidabile fino a 0,138N esempi, dove N è il numero di neuroni della rete.
Per l'addestramento delle reti di Hopfield si utilizzano delle particolari operazioni lineari (matriciali), e non è necessario ricorrere quindi ad algoritmi iterativi.
Per il funzionamento in avanti occorre invece calcolare iterativamente le uscite fino a quando esse non si stabilizzano su una soluzione.
Le reti di Kohonen (dette anche *Self-Organizing Maps*) invece costituiscono un esempio di reti in grado di apprendere senza bisogno di supervisione. In esse gli ingressi vengono forniti contemporaneamente a tutti i neuroni, ed i pesi vengono ritoccati in base a criteri di "somiglianza" e di "località", piuttosto che per diminuire l'errore dell'uscita.
La rete presenta la capacità di raggruppare automaticamente gli ingressi per somiglianza, e quindi di crearne automaticamente una classificazione semantica e anche topologica. Anche in questo caso vengono utilizzate formule apposite per l'addestramento.

CONCLUSIONI

Si conclude qui questa breve panoramica sul mondo delle reti neurali (argomento troppo vasto per essere trattato in maniera esauriente in un solo articolo!).
Nella prossima puntata vedremo più da vicino molti aspetti pratici ed alcuni esempi legati all'implementazione ed all'utilizzo delle reti neurali.

Bibliografia minima

1. S. Russel, P. Norvig - "Intelligenza Artificiale: Un Approccio Moderno", Prentice-Hall / UTET
2. Richard P. Lippmann, "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine, Aprile 1987, pag 4-22
3. Sito Internet: www.generation5.org
4. FAQ del News Group Usenet: comp.ai.neural-nets

Alimentatori switching:

In questa puntata inizieremo a trattare le caratteristiche del nostro "serbatoio" magnetico, l'induttanza. Presenterò poi una tipologia ibrida, cioè che deriva da due delle tipologie che già conoscete: la buck-boost. Con l'ausilio di un circuito integrato di nuovo tipo (PWM), realizzeremo infine il nostro classico circuito di test.

LE ORIGINI DEL MAGNETISMO

Riproduciamo un noto esperimento, visibile in figura 1, con l'ausilio di un filo percorso da una corrente elettrica e di alcune bussole tascabili. Nella parte a sinistra della figura 1, ad interruttore spento, notiamo che le piccole bussole (che non sono altro che magneti permanenti, liberi di muoversi attorno al proprio asse), si orientano tutte secondo il polo Nord terrestre. Facendo scorrere una corrente nel filo, tramite l'accensione dell'interruttore, le bussole si orienteranno secondo cerchi concentrici, aventi centro proprio nel punto di passaggio del filo. Questo esperimento dimostra che:

> qualsiasi corrente elettrica che scorre in un conduttore genera un campo magnetico

Ovviamente il campo magnetico generato avrà un'intensità ben definita e limitata, cioè bussole collocate oltre una certa distanza dal filo non risentiranno più del campo, e non si orienteranno più. L'intensità del campo magnetico si indica con il simbolo H, da non confondere con l'unità di misura dell'induttanza, l'Henry. Appare evidente, anche senza dimostrazione, che l'intensità del campo magnetico H dipende dalla corrente che scorre nel filo, ed infatti è direttamente proporzionale, cioè se la corrente raddoppia, anche H raddoppia. Inoltre se invece di un singolo filo, ne colleghiamo molti, **tutti attraversati dalla stessa corrente**, (in pratica, realizzando un solenoide) l'effetto aumenta proporzionalmente, come esemplificato in figura 2. Possiamo quindi scrivere che H è proporzionale sia alla corrente che al numero delle spire, secondo una costante di proporzionalità k:

$$H = k * I * N$$

(I corrente in Ampere, N numero di spire, k costante)

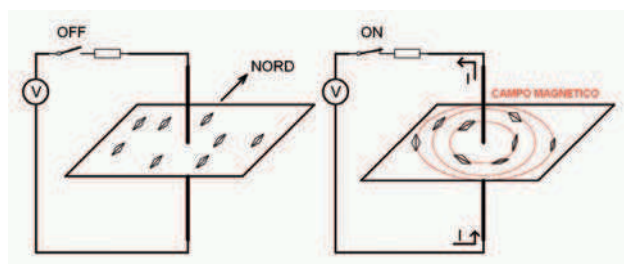



Figura 1 *Esperimento sul magnetismo*

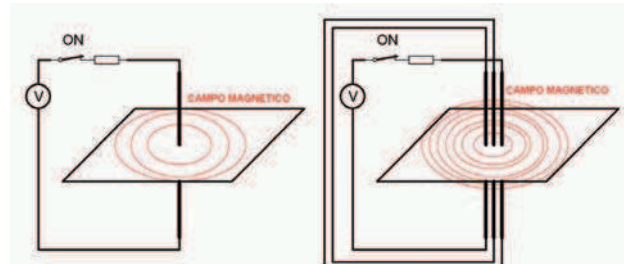



Figura 2 *Campo magnetico in un solenoide*

Il controllo PWM negli alimentatori switching

di Romano Bernarducci
r.bernarducci@farelettronica.com

L'intensità del campo magnetico H si misura in:

A / m
(A = Ampere, m = metro)

A volte si possono trovare per H, le unità di misura A * sp / m, Amperspira per metro, oppure A * T / m, dove T sta per l'inglese "turn", spira appunto. Essendo le spire un numero adimensionale, tutte le precedenti espressioni sono equivalenti.
Un'altra grandezza strettamente collegata all'intensità del campo magnetico H, è la cosiddetta densità di flusso, indicata con la lettera B. **In pratica la densità di flusso fornisce un'indicazione del numero di linee di campo magnetico che attraversano un'area unitaria**.
L'unità di misura corrispondente è il Tesla, indicato con la lettera T:

$$1 \text{ Tesla} = 1T = 1kg / (A * s^2) = 10000 \text{ Gauss}$$
(A = Ampere, s = secondo, kg = kilogrammo)

Il Tesla è l'unità di misura riconosciuta dal Sistema Internazionale SI, tuttavia non è raro trovare dati riferiti in Gauss, quindi ho indicato il fattore di conversione, pari stranamente a 10000 (e non 1000 come ci si potrebbe aspettare). Perchè ho detto che B e H sono strettamente collegate? Perchè, nel vuoto, e quindi approssimativamente nell'aria, vale la formula:

$$B = \mu_0 * H$$

con μ_0 costante pari a $4 * \pi * 10^{-6}$ [$kg * m / (A^2 * s^2)$], denominata *permeabilità magnetica assoluta.*
Per fare un esempio, applicando la formula precedente ed un'altra qui non riportata, si può calcolare che la densità di flusso generato da un filo percorso da 1A, a 1m di distanza, è pari a 200nT, cioè $200 * 10^{-9}$ Tesla.

Il campo magnetico terrestre è ben più elevato, e vale circa 60μT, cioè $60 * 10^{-6}$ Tesla = 0,6 Gauss.

Ricordo ora che gli elettroni degli atomi sono cariche in movimento (cioè una corrente), e quindi generano anch'essi un campo magnetico "interno".
Cosa succede se immergiamo un pezzo di un qualunque materiale nel campo magnetico prodotto da un solenoide? La risposta è, naturalmente: dipende dal materiale.
Per la maggior parte dei materiali come il rame, l'alluminio, la grafite, il legno, la plastica, vale ancora la relazione precedente $B = \mu_0 * H$, in quanto il campo magnetico "interno" si cancella a causa della struttura molecolare oppure per proprietà di simmetria dell'atomo. Per fortuna esistono alcuni materiali il cui campo magnetico "interno" non si annulla, anzi, si allinea con quello esterno, aumentandone notevolmente l'effetto: i materiali ferromagnetici.

I MATERIALI FERROMAGNETICI

Questi sono materiali tipicamente metallici, come il ferro, il nickel, il manganese. Il fattore di proporzionalità che lega la densità di flusso B al campo magnetico H, non è più la costante μ_0, ma..un disastro! Immaginando di eseguire un esperimento, aumentiamo gradualmente H da zero (per esempio aumentando la corrente che scorre nel solenoide), e grafichiamo il valore di B, misurato con un opportuno sensore, come in figura 3.
Si notano subito tre cose:

1. A parità di H, B è molto più elevato in questi materiali rispetto al vuoto.
2. La dipendenza da H non è lineare.
3. Oltre un certo valore di H, B non aumenta più (appiattimento della curva).

In quest'ultimo caso si dice che il materiale è **saturo**, cioè fisicamente tutti i campi magnetici "interni" si sono allineati a quello generato esternamente e non possono più contribuire all'incremento di flusso.
Cosa succede se adesso diminuiamo il campo magnetico H fino a zero?
Ci aspettiamo di ritornare al punto di partenza seguendo la stessa curva...invece no! La curva passa molto più in alto rispetto alla precedente. Se continuiamo a diminuire H sotto lo zero (cioè invertiamo la corrente), e poi lo riportiamo fino al massimo, otteniamo il grafico di figura 4.

I più esperti avranno sicuramente riconosciuto il classico *ciclo di isteresi di magnetizzazione*, che è tipica di tutti i materiali ferromagnetici. Per contrasto, la curva di figura 3 viene chiamata *curva di prima magnetizzazione o curva vergine*, ma non è di interesse per il nostro corso.
Esiste un'altra categoria di materiali che presentano caratteristiche ferromagnetiche (quindi ciclo di isteresi B-H ed alta densità di flusso B): **le ferriti**. Questi sono materiali ceramici ricavati per sinterizzazione, cioè per trattamento contemporaneo ad alta pressione ed alta temperatura, di polveri di ossidi di ferro, nichel, manganese e zinco.
Rispetto ai metalli ferromagnetici, hanno una conducibilità elettrica molto minore, fatto molto importante ai fini della diminuzione delle perdite, come vedremo nella prossima puntata.
NOTA: a rigore le ferriti sono materiali *ferrimagnetici* e non *ferromagnetici*, ma per la nostra trattazione i due tipi sono perfettamente equivalenti e quindi useremo sempre la dizione *ferromagnetici*.

PERDITE NELLE INDUTTANZE: PARTE 1

Come noto, qualsiasi induttanza viene realizzata tramite un avvolgimento di filo conduttore, tipicamente rame, su un opportuno supporto, che, nel caso degli alimentatori switching, è esclusivamente di materiale ferromagnetico. Per quanto si cerchi di utilizzare un filo conduttore di sezione elevata e quindi di bassa resistenza, quest'ultima va sempre tenuta in considerazione nel calcolo delle perdite. In pratica l'induttanza reale ha lo schema equivalente (semplificato) di figura 5.
Il calcolo delle perdite elettriche è semplicemente dato dalla legge di dissipazione di potenza sulle resistenze:

$$P_{DISS,L} = R_L * I_{L,AVG}^2$$

dove $I_{L,AVG}$ indica la corrente media nell'induttanza (o quella efficace RMS per i più precisi). Nella prossima puntata esamineremo le altre fonti di

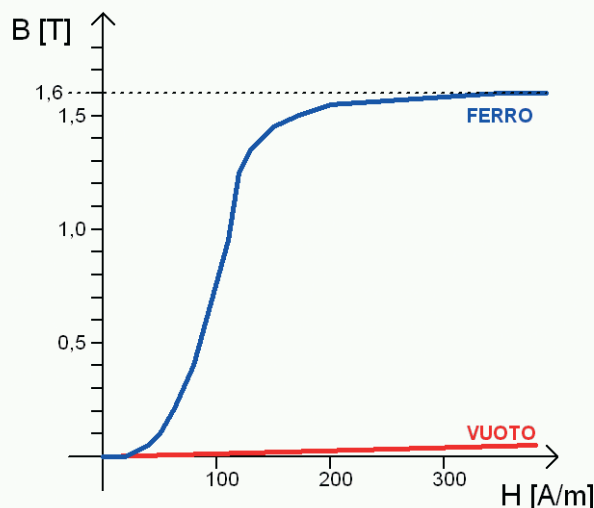



Figura 3 Grafico B-H per un materiale ferromagnetico (ferro)

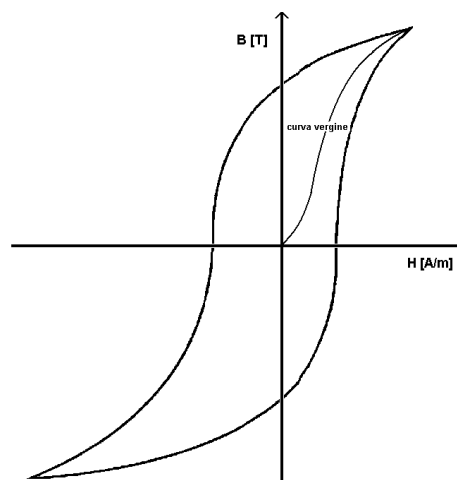



Figura 4 Ciclo di magnetizzazione B-H per un materiale ferromagnetico

FORTRONIC

ELECTRONICS FORUM

components - production & new solutions

INNOVATION · PRODUCTION · DISTRIBUTION

21-23 Aprile 2005

Fiera Padova - www.fortronic.it

INNOVATION DAY

da Giovedì 21

GLI INCONTRI DELL'INNOVAZIONE

Università, centri di ricerca, spin-off, progettisti

➤ **Sensori Wireless** — *convegno*
Stato dell'arte e tendenze per le reti wireless di sensori.
Focus su tecnologie emergenti *(Zigbee e Bluetooth)*.

➤ **Casa intelligente** — *area demo*
in collaborazione con CNA e Ordine degli Architetti PD
Utilizzo dei sensori wireless nella domotica:
trasmissione, identificazione e sicurezza (RFID).

➤ **Sistemi di visione** — *area demo & convegni*
Dall'automazione alla sicurezza: le opportunità
applicative proposte da Imprese e Spin off universitari.

➤ **ASIC strutturati e FPGA** — *convegno*
Prestazioni, costi ridotti contro time-to-market e flessibilità:
due tecnologie a confronto per le soluzioni migliori.

➤ **VI Design-In Award**
Premio alle capacità progettuali nella automazione
industriale con la "wireless sensor technology".

PRODUCTION DAY

Venerdì 22

GLI INCONTRI DELLA PRODUZIONE

Contract Manufacturers, Ems, Odm, tecnici

➤ **Lead Free** — *convegno*
I cambiamenti richiesti nella filiera produttiva:
dall'ingegnerizzazione al testing. L'introduzione
delle normative ROHS su componenti e materiali.

➤ **Linea di produzione "senza piombo"** — *area demo*
Una verifica 'on line' su come affrontare correttamente
i vincoli del lead free.

➤ **SEP & RAEE** — *area convegno con Sep-Fiera PD*
Attrezzature e interventi per il trattamento
e il recupero del Rifiuto Elettronici.

➤ **Outsourcing** — *tavola rotonda*
"Produrre in Italia o all'estero.
Quando e come scegliere!".

➤ **Assodel Gala e Award**

DISTRIBUTION DAY

fino a Sabato 23

GLI INCONTRI DEL MERCATO

Fornitori, clienti, distributori, buyers

➤ **XII International Electronics Distribution Meeting**
La "Demand creation" nei cambiamenti
di un'economia globale.
Prospettive e trend della distribuzione.

IDEA

➤ **Il Buyer nella Electronic Community**
Come equilibrare informazione e flessibilità
con logistica e ottimizzazione dei costi.

➤ **Europartner Forum** — *tavola rotonda*
Sessione di marketing internazionale dedicata
agli European Distribution Mangers.

➤ **Trading Post** — *area demo*
Ricerca di rappresentanze e rappresentate.
Collaborazioni con associazioni straniere.
Delegazioni e missioni

iscriviti su WWW.FORTRONIC.IT riceverai gratis la tessera d'ingresso e un abbonamento alle riviste dell'electronic community

perdita di energia nelle induttanze.

TIPOLOGIA BUCK-BOOST

Questa tipologia è un ibrido di due tipologie a voi ben note, già presentate nelle scorse puntate del corso: la buck e la boost, ovviamente. La figura 6 mostra lo schema di principio.

In effetti sono proprio due alimentatori switching buck e boost in serie, con "l'ottimizzazione" consistente nell'usare la stessa induttanza. Un momento...non vi sembra inutile abbassare la tensione di ingresso, con la parte buck a sinistra, per poi elevarla di nuovo con la parte boost di destra? Riflettete in quale occasione questo può tornare utile...ma si, **quando la tensione di ingresso può essere sia inferiore che superiore a quella di uscita!** Supponiamo infatti di avere a disposizione una tensione di ingresso variabile da +9V a +18V, da cui si desideri ottenere una tensione di uscita costante ed uguale a +12V.
Utilizzando la tipologia buck-boost, possiamo usare la parte "buck" per abbassare la tensione di ingresso al valore +8V fisso, e poi utilizzare la parte "boost" per aumentare quest'ultima a +12V **(in realtà in nessun punto del circuito troveremo +8V, ma è solo un esempio spero illuminante)**.

IDEALE REALE

Figura 5 *Circuito equivalente (I) per l'induttanza*

Ricorderete che nella puntata precedente avevo presentato la tipologia inverting che fornisce lo stesso risultato, a scapito però dell'inversione di polarità tra ingresso e uscita. La tabella 1 fornisce una comparazione delle caratteristiche tra queste due tipologie.

Analizzando la tabella 1 e la figura 6, è immediato constatare che il maggior costo e la minore efficienza della tipologia buck-boost è legato al numero doppio di interruttori presenti, tuttavia è l'unica **(per ora)** utilizzabile se si vuole che ingresso ed uscita condividano la stessa massa elettrica.
Nella figura 6, per confronto con le tipologie già note, è immediato inserire diodi (interruttori non comandati) nelle posizioni 2a e 2b, e MOSFET a canale P e N rispettivamente nelle posizioni 1a e 1b, come in figura 7.

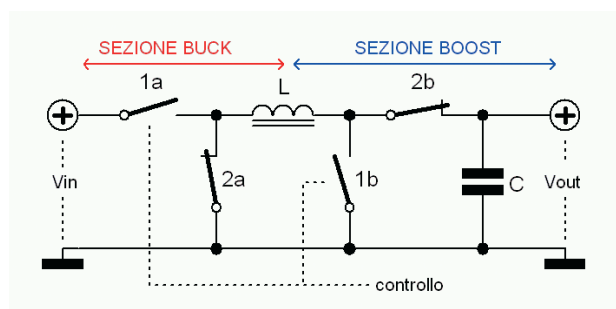



Figura 6: *Schema di principio (I) di un alimentatore switching di tipo buck-boost*

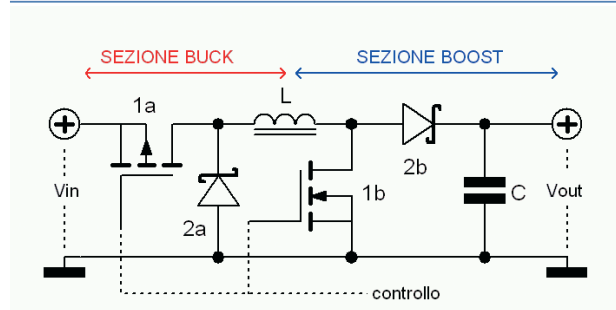



Figura 7 *Schema di principio (II) di un alimentatore switching di tipo buck-boost*

Tipologia	V_{IN} può essere sia > che < $\lvert V_{OUT}\rvert$?	Inversione del GND?	Complessità (costo)	Efficienza teorica η
buck-boost	SI	NO	MAGGIORE	MINORE
inverting	SI	SI	MINORE	MAGGIORE

Tabella 1 *Comparazione tra tipologie*

Il controllo agisce su entrambi i MOSFET contemporaneamente durante la fase ON, mentre durante la fase OFF, l'immediata inversione di polarità ai capi di L fa si che l'energia magnetica accumulata, fluisca verso il carico attraverso i diodi 2a e 2b. In figura 8 sono riportati i grafici delle correnti nei vari componenti.

Un piccolo problema nasce dal fatto che per attivare il PMOS 1a occorre fornire una tensione di Gate negativa rispetto al Source che è a V_{IN} (ad esempio 0V), mentre per attivare l'NMOS 1b occorre una tensione positiva rispetto al Source che è a massa (ad esempio 10V). Il controllo dovrebbe quindi fornire due uscite complementari per pilotare correttamente i due MOS, oppure è necessario prevedere un inverter esterno tra i due Gate, oppure ancora... si può connettere il Gate dell'NMOS 1b come in figura 9.

In questo caso il controllo gestisce solo il PMOS 1a, il quale una volta attivato, fornisce V_{IN} sul suo Drain, che è connesso al Gate dell'NMOS 1b, e si attiva quindi quasi contemporaneamente.

TIPOLOGIA DI CONTROLLO PWM

Introduciamo finalmente la tipologia di controllo PWM. Come tutti saprete, PWM è l'acronimo di Pulse Width Modulation, traducibile come Modulazione a Larghezza di Impulso.
In pratica un controllo PWM varia l'ampiezza dell'impulso di uscita, che rappresenta la grandezza CONTROLLANTE, a seconda dell'errore tra la grandezza CONTROLLATA ed un riferimento che rappresenta il valore desiderato (figura 10).

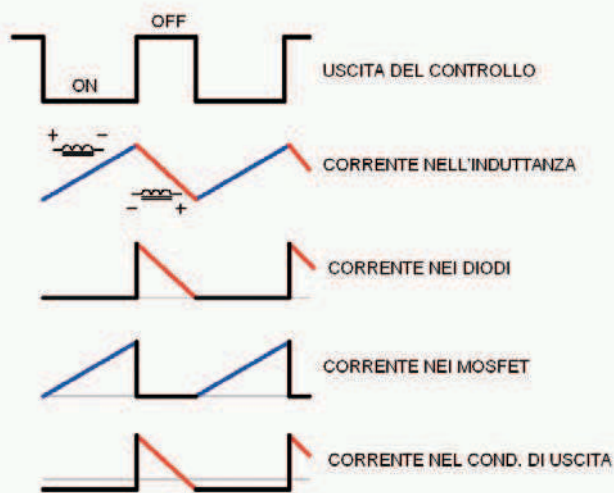



Figura 8 Correnti tipiche in un alimentatore switching buck-boost

È questa la tipologia di controllo PWM Voltage-Mode, in quanto l'uscita del circuito di controllo dipende solo dalla tensione di uscita e non da altre grandezze. La tipologia PWM Voltage-Mode è stata la prima, storicamente, ad essere introdotta e risente di alcuni problemi cui accenneremo più avanti; tuttavia i circuiti di controllo integrati che la implementano sono molto economici, facilmente reperibili ed in grado di funzionare egregiamente in moltissime applicazioni.

UN CIRCUITO INTEGRATO PWM V-MODE: TL5001

Un circuito che integra un controllo PWM V-Mode è il TL5001 della Texas Instruments (www.ti.com). Viene fornito in un piccolo packa-

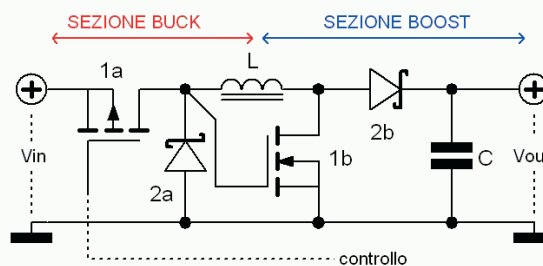



Figura 9 Schema di principio (III) di un alimentatore switching di tipo buck-boost

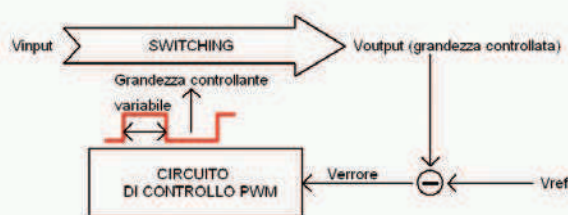



Figura 10 Schema di principio di un controllo PWM

ge a 8 pin, sia SMD che DIP, ed è facilmente reperibile su RS (www.rs-components.it) e, probabilmente, anche attraverso i normali canali di rivendita di materiale elettronico. In figura 11 è riportata la struttura interna semplificata dell'integrato.

Tralasciando il pin 5, SCP, che implementa la protezione dai corto circuiti e che non useremo per semplicità, l'interno del TL5001 consiste, come al solito, di:

- Un oscillatore triangolare a frequenza programmabile tramite una resistenza sul pin 7, RT.
- Un riferimento di tensione a 1V (trascurate quello a 2,5V).
- Un amplificatore di errore, cioè un classico op-amp.
- Un comparatore.
- Uno stadio di uscita a transistor NPN, open-collector.
- Un controllo del duty-cycle massimo tramite una resistenza sul pin 6, DTC.

Vedremo più avanti la funzione dell'UVLO.
Il cuore del generatore PWM è proprio il comparatore; infatti se supponiamo di graficare la sua tensione di uscita (collegata all'AND e quindi al transistor di uscita), per vari valori di tensione di ingresso sul pin 3 otteniamo quanto visibile in figura 12.

È evidente quindi che la durata dell'impulso di uscita è direttamente proporzionale alla tensione sul pin 3, cioè alla tensione di errore presente all'uscita dell'op-amp. Maggiore è l'errore, ad esempio a causa di un improvviso sovraccarico, maggiore è l'ampiezza dell'impulso di uscita, che pilota la parte di potenza dello switching ed aumenta l'energia trasferita ad ogni ciclo tramite il serbatoio magnetico.

COS'È L'UVLO?

L'UVLO è un acronimo per *Under-Voltage-Lock-Out*, traducibile con un poco di fantasia in *Blocco di Bassa Tensione*. **Il suo scopo è quello di spegnere lo switching se la tensione di ingresso è troppo bassa**. Sapete dire perchè?
Ricordate che qualsiasi alimentatore switching è un dispositivo a POTENZA COSTANTE?
Per fare un esempio, se il carico assorbe 100W, la potenza assorbita dall'ingresso sarà 100W + le perdite, cioè supponiamo 120W in totale.
Se lo switching è alimentato, diciamo a 12V, questo si traduce in un assorbimento di 120 / 12 = 10A. Se la tensione di ingresso si dimezzasse a 6V, la corrente assorbita salirebbe a ben 120 / 6 = 20A, e addirittura a 40A se la tensione scendesse ipoteticamente a 3V!. È evidente che il dimensionamento del MOSFET, della bobina, dei diodi, è stato effettuato per una certa corrente operativa massima...quadruplicare questo valore vuol dire bruciare il fusibile, se siete fortunati, o qualche costoso componente attivo se non lo siete! Per riepilogare:

> Se la tensione di ingresso non è sufficiente per il corretto funzionamento, l'UVLO protegge lo switching, spegnendolo o ritardandone l'accensione.

L'UVLO presente nel TL5001 non è programmabile, ma interviene se la tensione di ingresso scende sotto i 2,9V, con ±100mV di isteresi. Circuiti integrati più evoluti (ma con più piedini!) possono permettere la regolazione del livello di intervento dell'UVLO.

ALIMENTATORE BUCK-BOOST

Per l'applicazione pratica con il PWM TL5001 utilizzeremo i valori seguenti.
Parametri di progetto desiderati:
$V_{IN,MIN}$ = +9V = Tensione di ingresso minima
$V_{IN,MAX}$ = +18V = Tensione di ingresso massima

È chiaro che la protezione UVLO del TL5001, fissa a 2,9V, è d'aiuto solo nel momento di accensione dello switching, ma non durante il funzionamento normale.

La tensione di uscita desiderata è:
V_{OUT} = +12V con una corrente massima

$$I_{OUT,MAX} = 500mA$$

PASSO 1 - Calcolo della massima potenza di uscita

$$P_{OUT,MAX} = |V_{OUT}| * I_{OUT,MAX}$$

nel nostro caso:

$$P_{OUT,MAX} = 12 * 0,500 = 6W$$

PASSO 2 - Stima dell'efficienza e calcolo della massima potenza di ingresso

Come al solito sceglieremo un valore a piacere compreso tra il 70 e 90%.
Dalla nota formula:

$$P_{IN} = P_{OUT} / \eta * 100$$

Scegliendo η = 75%, otteniamo:

$$P_{IN,MAX} = 6 / 75 * 100 = 8W$$

PASSO 3 - Calcolo della massima corrente di ingresso

Per il calcolo occorre utilizzare il valore minimo della tensione di ingresso, che rappresenta il caso peggiore per la corrente di ingresso:

$$I_{IN,MAX} = P_{IN,MAX} / V_{IN,MIN}$$

nel nostro caso:

$$I_{IN,MAX} = 8 / 9 = 0,889A$$

Questo valore è utile per dimensionare il fusibile di protezione all'ingresso, di tipo ritardato come ovvio.

PASSO 4 – Calcolo della corrente nell'induttanza

Figura 11 Struttura interna semplificata dell'integrato TL5001

Anche per la tipologia buck-boost, la corrente nell'induttanza non coincide con quella di ingresso, e neanche con quella che scorre nel carico, come si può facilmente comprendere dalla figura 8, ingrandita in figura 13.
La formula 1, che fornisce la massima corrente nell'induttanza, è identica quella già vista nella precedente puntata per la tipologia inverting.

Formula 1
$I_{L,MAX} = 2 * I_{OUT} * (t_{ON} / t_{OFF} + 1)$,
o, equivalentemente:
$I_{L,MAX} = 2 * I_{OUT} / (1 - \text{duty-cycle})$

Figura 12 Generatore PWM

PASSO 6 - Calcolo del duty-cycle

Calcoliamo il duty-cycle, cioè il periodo ON rispetto al totale del ciclo (ON+OFF), usando la nota formula:

$$|V_{L,ON}| * t_{ON} = |V_{L,OFF}| * t_{OFF}$$

$V_{L,ON}$ è la tensione ai capi dell'induttanza durante la fase ON. Purtroppo non sappiamo ancora quale sarà il duty-cycle, quindi

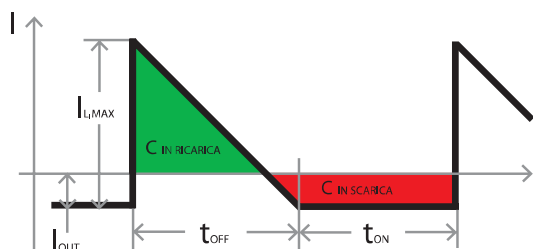



Figura 13 *Corrente nel condensatore di uscita per tipologia buck-boost*

non conosciamo la corrente $I_{L,MAX}$ nell'induttanza e non possiamo calcolare la caduta di tensione sulle $r_{DS,ON}$ dei MOSFET.

Supponiamo quindi trascurabile la caduta di tensione sui MOSFET ed eseguiamo i calcoli: alla fine verificheremo la caduta di tensione sui MOSFET prescelti e, se necessario, torneremo a questo punto per affinare i calcoli. Alla tensione di ingresso minima, risulta quindi:

$$V_{L,ON} = V_{IN,MIN} = 9V$$

$V_{L,OFF}$ è la tensione ai capi dell'induttanza durante la fase OFF. Poiché utilizzeremo due diodi schottky del tipo 1N5818 come interruttori 2a e 2b, la caduta di tensione complessiva sarà pari a $2 * V_D = 2 * 0{,}5V = 1V$.

$$|V_{L,OFF}| = V_{OUT} + 2 * V_D = 12 + 1 = 13V$$

da cui:

$$9 * t_{ON} = 13 * t_{OFF}$$
$$t_{ON} = 1{,}444 * t_{OFF}$$

e quindi, per il duty-cycle massimo:

$$\text{duty-cycle}_{MAX} = t_{ON} / (t_{OFF} + t_{ON}) = 59{,}1\%$$

Ripetendo i medesimi calcoli anche per $V_{L,ON} = V_{IN,MAX} = 18V$ otterremo il duty-cycle minimo:

$$\text{duty-cycle}_{MIN} = 41{,}9\%$$

Sostituendo i duty-cycle appena calcolati nella formula 1 otteniamo per la corrente max. nell'induttanza:

$$I_{L,MAX} = 2 * I_{OUT} / (1 - \text{duty-cycle}_{MIN}) =$$
$$= 2 * 0{,}500 / (1 - 0{,}419) = 1{,}721A$$
$$I_{L,MAX} = 2 * I_{OUT} / (1 - \text{duty-cycle}_{MAX}) =$$
$$=2 * 0{,}500 / (1 - 0{,}591) = 2{,}445A$$

Induttanza L in μH	Frequenza di Switching f_{SW}
22	98,9kHz
47	46,3kHz
100	21,8kHz

Tabella 2 *Coppie frequenza-induttanza*

Dai valori di $I_{L,MAX}$ è immediato calcolare il valore dell'induttanza e della frequenza di switching. Vale infatti la relazione, già dimostrata in precedenza:

$$f_{sw} * L = (V_{L,ON} * \text{duty-cycle}) / I_{L,MAX}$$

Come al solito l'$I_{L,MAX}$ da utilizzare è quella più alta, che corrisponde al duty-cycle maggiore e alla tensione minima di ingresso.

La Tabella 2 riporta le frequenze risultanti per vari valori dell'induttanza L.

Questa volta ho utilizzato un'induttanza da 50μH-3A-0,08Ω, valore un poco fuori standard ma reperibile sul catalogo Distrelec (codice 35 07 84). In questo caso la frequenza di switching dovrà essere circa pari a 43,5kHz.

PASSO 7 – Selezione del MOSFET

La V_{DSS} dei MOSFET deve essere in valore assoluto, almeno pari a 20V, ma come sapete questo è un parametro rispettato praticamente da tutti i MOSFET. Per quanto riguarda la corrente, applicando la regola di utilizzare un MOSFET con **$I_{D,MAX}$ a 25°C da 2 a 4 volte superiore alla massima corrente prevista**, abbiamo bisogno di un componente con ID compresa tra 5 e 10A. Come prima scelta, proviamo i già noti:

IRF9520, PMOS 6,8A-100V-0,6Ω
IRF620, NMOS 5,2A-200V-0,8Ω

Durante la fase ON, la $r_{DS,ON}$ totale dei due MOS in serie sarà pari a 0,6 + 0,8 = 1,4Ω. Questa, moltiplicata per la corrente di picco, causerebbe ben 3,5V di caduta di tensione!

Quindi, le sole cadute resistive sui MOSFET si "mangerebbero" quasi il 40% della tensione di ingresso, con negative ripercussioni sull'efficienza generale e sul calore dissipato dai MOSFET. In questo caso, ed in genere quando $I_{L,MAX} >> I_{OUT}$, conviene indirizzarsi sui grossi calibri, in quanto le perdite dovute alle maggiori capacità di giunzione saranno ampiamente compensate dalla

minore $r_{DS,ON}$. Scegliamo quindi:

IRF9530, PMOS 14A-100V-0,2Ω
IRF540, NMOS 19A-55V-0,06Ω

con i quali otteniamo una rDS,ON totale pari a 0,26ø, con conseguente, ridotta caduta di tensione pari a circa 0,6V. Da notare infine che sui Gate dei MOSFET è presente una tensione massima di circa 18V. La maggior parte dei MOSFET di potenza resiste a tensioni di Gate fino a ±20V, quindi non sono necessarie particolari protezioni. Tuttavia, nei vostri progetti, tenete bene a mente questi limiti, ed eventualmente inserite degli zener nello stadio di pilotaggio o direttamente sui Gate.

PASSO 8 - stima delle perdite dissipative

Come già ampiamente detto, non è facile calcolare il massimo della **somma** delle perdite MOSFET+DIODI, in quanto dipendono dal duty-cycle.
Calcoliamo quindi le perdite nei casi limite della tensione di ingresso, 9V e 18V, approssimando la corrente media come metà di quella di picco:

PERDITE NEI MOSFET (V_{IN} = 9V)

$$P_{DISS,MOSFET} = r_{DS,ON} * (I_{PK} / 2)^2 * \text{duty-cycle} = 0,260 * (2,445 / 2)^2 * 59,1\% = 0,230W$$

PERDITE NEI DIODI (V_{IN} = 9V)

$$P_{DISS,D} = 2 * V_D * (I_{PK} / 2) * \text{duty-cycle}_{(D)} = 2 * 0,5 * (2,445 / 2) * (1 - 0,591) = 0,500W$$

PERDITE RES. INDUTTANZA (V_{IN} = 9V)

$$P_{DISS,L} = (I_{PK} / 2)^2 * R_L = (2,445 / 2)^2 * 0,08 = 0,120W$$

Per un totale di:

$$P_{DISS,TOT} = P_{DISS,MOSFET} + P_{DISS,D} + P_{DISS,L} = 0,230 + 0,500 + 0,120 = \mathbf{0,850W\ (VIN = 9V)}$$

PERDITE NEI MOSFET (V_{IN} = 18V)

$$P_{DISS,MOSFET} = r_{DS,ON} * (I_{PK} / 2)^2 * \text{duty-cycle} = 0,260 * (1,721 / 2)^2 * 41,9\% = 0,081W$$

PERDITE NEI DIODI (V_{IN} = 18V)

$$P_{DISS,D} = 2 * V_D * (I_{PK} / 2) * \text{duty-cycle}_{(D)} = 2 * 0,5 * (1,721 / 2) * (1 - 0,419) = 0,500W$$

PERDITE RES. INDUTTANZA (V_{IN} = 18V)

$$P_{DISS,L} = (IPK / 2)^2 * R_L = (1,721 / 2)^2 * 0,08 = 0,059W$$

Per un totale di:

$$P_{DISS,TOT} = P_{DISS,MOSFET} + P_{DISS,D} + P_{DISS,L} = 0,081 + 0,500 + 0,059 = \mathbf{0,637W\ (VIN = 18V)}$$

PASSO 10 - calcolo del condensatore di uscita

Sappiamo che il condensatore di uscita influisce sul ripple di tensione di uscita, tramite la sua **capacità** e la sua **ESR**. A svantaggio della tipologia buck-boost, c'è il fatto che la corrente nel condensatore è triangolare, con un tratto verticale più difficile da filtrare.
La procedura di calcolo del condensatore di uscita è identica a quella già vista per la tipologia inverting:

1. Calcolare l'ESR con la **ESR = ΔVESR / ΔI**, ipotizzando ΔV_{ESR} pari al 90% del totale.
2. Scegliere un condensatore di uscita che abbia un ESR inferiore a quello calcolato, o mettere in parallelo due o più condensatori con ESR maggiore.
3. (Opzionale ma consigliabile per l'affidabilità) calcolare il valore efficace del ripple di corrente **triangolare** con la formula 2 e verificare che sia inferiore al massimo sopportabile dal condensatore scelto nella fase 2.

Formula 2

$$I_{RMS} = [I_{PK} * (I_{PK} / 3 - I_{OUT}) * = (1 – \text{duty-cyle}) + I_{OUT}^2]^{0,5}$$

4. Calcolare il ripple di tensione dovuto alla capacità, ΔV_C, con la formula 3 e verificare che, sommato a quello del passo 1, il ripple totale sia accettabile. In caso contrario ripetere dal passo 1 ipotizzando una diversa percentuale per ΔV_{ESR}.

Formula 3

$$\Delta V_C = (I_{PK} - I_{OUT})^2 * t_{OFF} / (2 * C * I_{PK})$$

Nel nostro caso, **utilizzando per il calcolo la corrente di picco nel caso peggiore** (V_{IN} = 9V), ed ipotizzando un ripple totale di 100mVpp, otteniamo:

$$ESR = \Delta V / \Delta I = 0,1 * 90\% / 2,445 = 0,041\Omega$$

Scegliamo un condensatore Rubycon serie ZL, 105°C, da 470µF/50V (codice Distrelec 80 18 54), che esibisce un'ESR pari a 0,027Ω ed un ripple di *corrente* massimo pari a 2,05A. In questo caso il ripple di tensione dovuto all'ESR sarà:

$$\Delta V_{ESR} = \Delta I * ESR = 2,445 * 0,027 = 66mVpp$$

Il valore efficace del ripple di corrente è pari a:

$$I_{RMS} = [I_{PK} * (I_{PK} / 3 - I_{OUT}) * (1 - \text{duty-cyle}) + I_{OUT}^2]^{0,5} =$$
$$= [2,445 * (0,815 - 0,5) * 0,409 + 0,5^2]^{0,5} =$$
$$= 0,782A$$

abbondantemente minore del massimo, a garanzia di un ridotto auto-riscaldamento del condensatore e quindi di una sua maggiore durata. Calcoliamo il ripple di *tensione* dovuto alla capacità:

$$\Delta V_C = (I_{PK} - I_{OUT})^2 * t_{OFF} / (2 * C * I_{PK}) =$$
$$= (2,445 - 0,5)^2 * 9,4e\text{-}6 / (2 * 470e\text{-}6 * 2,445) =$$
$$= 15,5mVpp$$

Sommando il ripple di *tensione* dovuto ai due termini, si ottiene:

$$\Delta V_{TOT} = \Delta V_{ESR} + \Delta V_C = 66 + 15,5 = 81,5mVpp$$

che consideriamo accettabile.

A rigore, i due contributi al ripple totale non sono in fase, cioè quando il primo termine dovuto all'ESR è al massimo l'altro **non** è al massimo, tuttavia si può utilizzare cautelativamente il valore calcolato.

SWITCHING BUCK-BOOST CON CONTROLLER PWM TL5001

Lo schema completo è riportato in figura 14. Trascurate per ora i componenti R1-C3-C4-R4-C5. La resistenza R6 determina la frequenza dell'oscillatore interno al circuito integrato, che dobbiamo imporre a circa 43,5kHz, da cui il valore di circa 220k (vedi data-sheet del TL5001).

La resistenza R5 limita il duty-cyle massimo, e useremo questa possibilità come semplice protezione contro i cortocircuiti. Poiché abbiamo calcolato che il duty-cycle massimo è pari al 59%, utilizziamo per sicurezza il valore limite del 70% e calcoliamo la R5 tramite la complessa formula presente nel datas-heet del TL5001.

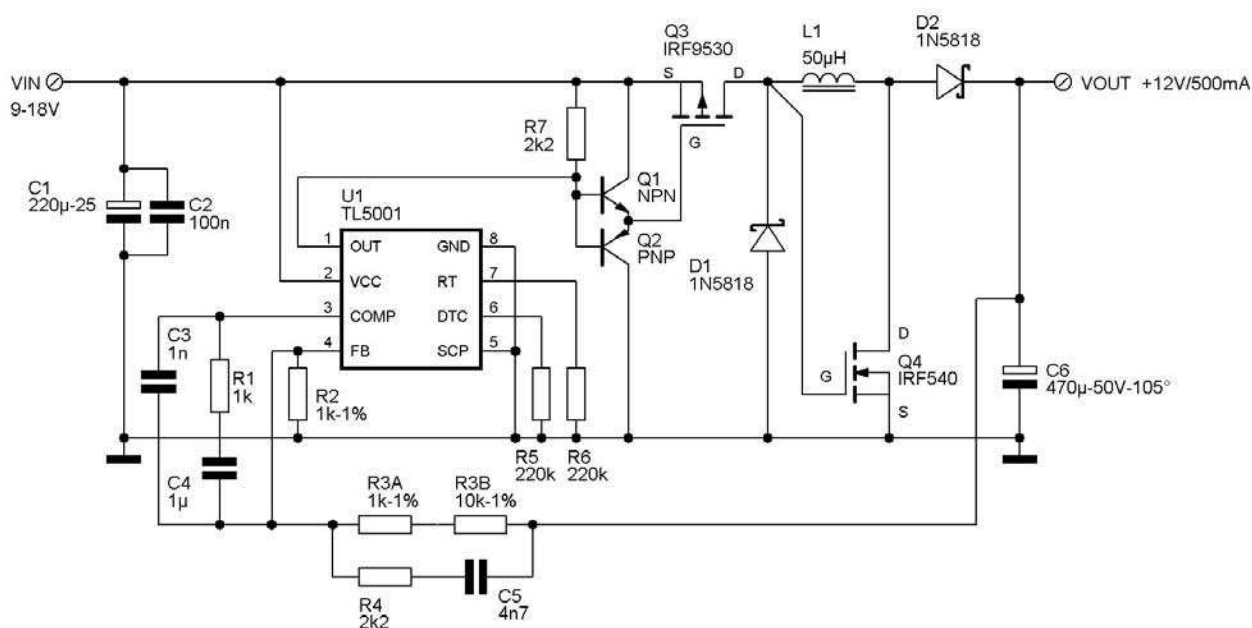



Figura 14 *Schema elettrico completo dell'alimentatore buck-boost +9..+18V -> +12V, 0,5A, 6W*

In genere si utilizzano valori identici o leggermente superiori a R6.
Per pilotare efficacemente il MOSFET Q3, utilizzeremo il solito push-pull inseguitore di emettitore esterno, quindi la corrente richiesta al TL5001 sarà molto bassa.
Per determinare i valori delle resistenze che regolano la tensione di uscita, ho calcolato prima il valore dell'attenuazione da applicare alla tensione di uscita per eguagliarla a quella del riferimento interno del TL5001, che vale 1V:

$$A_{TT} = V_{OUT} / V_{REF} = 12 / 1 = 12$$

Ricordando che in un partitore resistivo costituito da due resistenze Ra e Rb, l'attenuazione ai capi di Rb è pari a 1 + Ra / Rb, ed eguagliando alla formula precedente si ottiene:

1 + Ra / Rb = 12, da cui **Ra / Rb = 11**

cioè occorrono due resistenze il cui rapporto sia esattamente 1 a 11. Ho scelto i valori di 1k per Rb = R2, e di 11k per Ra = R3A+R3B, con tolleranza 1% onde evitare di ricorrere ad un piccolo trimmer.
A questo proposito ricordo che il TL5001 "standard" presenta una tolleranza pari a ±5% sul Vref interno (e quindi sulla precisione della tensione di uscita), ma esiste la versione con suffisso "A", il TL5001A con tolleranza ridotta al ±3%. I componenti R1-C3-C4-R4-C5 hanno il compito di modellare la risposta in frequenza e fase dell'anello di reazione che controlla la tensione di uscita, in modo da soddisfare due aspetti contrastanti: la stabilità dell'alimentatore e la prontezza di risposta alle variazioni di tensione. Purtroppo questo argomento è vasto e complesso, per cui non sarà trattato in questa sede. Chi volesse approfondire può trovare una buona trattazione, per le tipologie buck e boost, nelle Application Notes **slvu004.pdf e slvu005.pdf**, sempre sul sito della Texas Instruments.

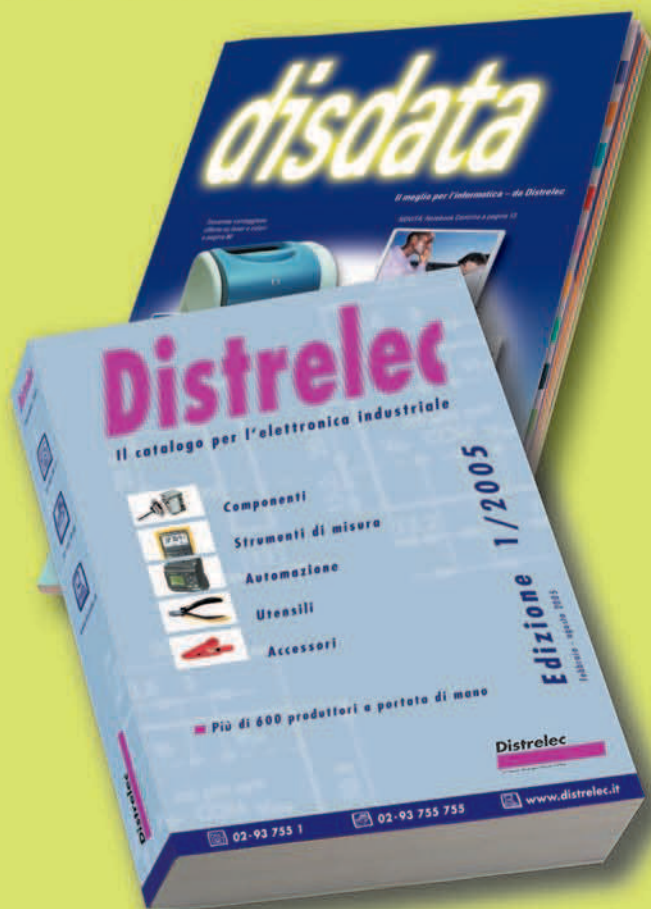
MISURE

Montato il circuito, e verificatone il corretto funzionamento, ho collegato in uscita un carico costituito da n.4 resistenze da 10Ω-2W in serie/parallelo, in modo da ottenere 25Ω, corrispondente praticamente al carico massimo. Senza carico esterno l'assorbimento del circuito è risultato essere pari a 5mA. Variando la tensione di alimentazione da 9 a 18V, ho ottenuto i risultati presentati in tabella (compresi i 5mA assorbiti a vuoto).
Dalla tabella notiamo che l'efficienza è piuttosto bassa, e dei circa 8W assorbiti in ingresso, 2W vengono dissipati in perdite.
Nel passo 8 abbiamo calcolato le perdite intorno a 0,6-0,8W. Secondo voi chi causa le restanti? Vi lascio con questo piccolo problemino (in realtà la soluzione è già stata citata).

CONCLUSIONI

Terminiamo con questa puntata l'utilizzo del TL5001. Nella prossima puntata introdurremo un nuovo componente PWM V-Mode, l'SG3525, con l'introduzione delle tipologie **push-pull**, e **forward**.
Proseguiremo poi nello studio dei materiali magnetici, ed introdurremo il trasformatore.

Tensione di ingresso [V]	Corrente assorbita [A]	Potenza assorbita [W]	Tensione di uscita [V]	Potenza fornita al carico (25Ω) [W]	Efficienza η [%]
9,0	0,870	7,83	11,99	5,75	73,4
12,0	0,650	7,80	12,00	5,76	73,8
15,0	0,540	8,10	12,00	5,76	71,1
18,0	0,460	8,28	12,01	5,77	69,7

Tabella 3 *Misure di efficienza sul prototipo*

Praticamente...

I Fotoaccoppiatori:

I fotoaccoppiatori consentono la trasmissione di segnali tra diverse sezioni di circuito elettricamente isolate tra loro. Questo li rende particolarmente adatti per tutte quelle applicazioni in cui è richiesto un alto grado isolamento. Questo mese vedremo come usarli realizzando un avvisatore ottico di chiamata telefonica come esempio applicativo.

ANALISI E SPECIFICHE DEL PROBLEMA

In ambienti particolarmente rumorosi può essere difficoltoso udire lo squillo del telefono. Esistono poi ambienti in cui il rumore deve essere ridotto al minimo ed in questo caso la suoneria del telefono potrebbe essere un disturbo. Il circuito da realizzare dovrà essere in grado di accendere una lampada 220Vac all'arrivo di una chiamata telefonica indipendentemente dal tipo suoneria impostata sull'apparecchio telefonico.

LA SOLUZIONE PROPOSTA

La figura 1 riporta lo schema elettrico richiesto dalle specifiche.
Lo squillo di una chiamata corrisponde, sulla linea telefonica, ad un'onda sinusoidale di ampiezza circa 80V e frequenza 25Hz.
Due squilli successivi sono separati da un silenzio di 4 secondi. Il circuito di figura 1 utilizza questa tensione per eccitare un triac ed azionare quindi una lampada collegata alla rete elettrica. Il fotoaccoppiatore risulta fondamentale per isolare le parti di circuito relativa alla linea telefonica e alla rete elettrica. Il condensatore C1 fa sì che solo il segnale in

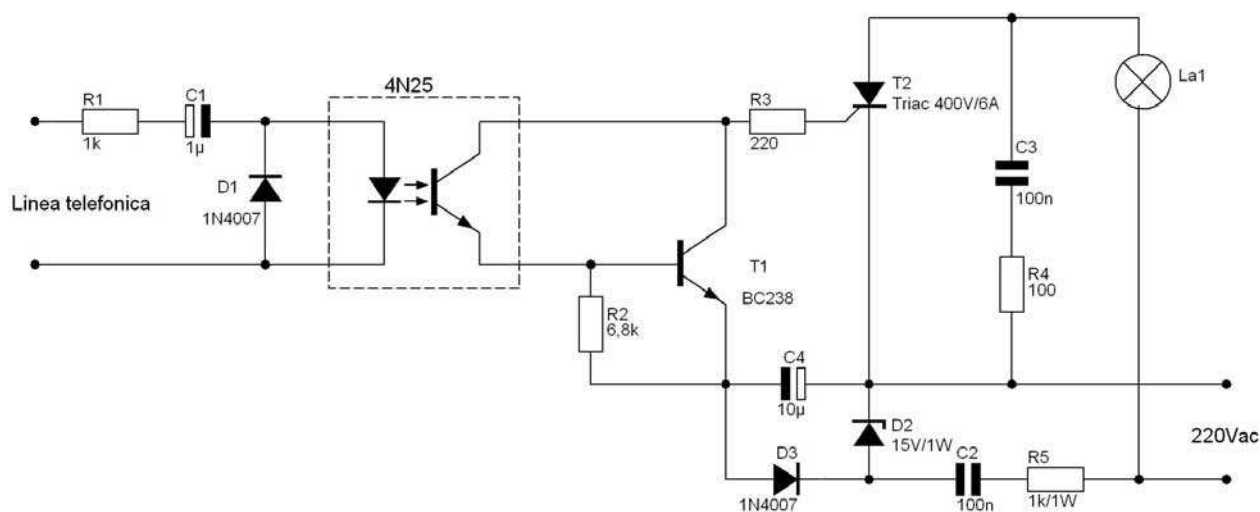



Figura 1 La soluzione proposta

di Maurizio Del Corso
m.delcorso@farelettronica.com

Avvisatore ottico di chiamata telefonica

alternata raggiunga il fotoaccoppiatore, R1 ne limita la corrente, mentre D1 garantisce che al fotoaccopiatore non giungano tensioni negative elevate (una tensione negativa manda in conduzione D1 che limita a circa 0,6V la tensione negativa ai capi del diodo emittente).
Quando il transistor interno al fotoaccoppiatore entra in conduzione, manda in conduzione anche T1 il quale fornisce la corrente necessaria all'eccitazione del triac.
L'alimentazione di T1 è ottenuta direttamente dalla rete a 220V grazie alla resistenza R5 ed il condensatore C2 che ne diminuiscono il valore massimo. La tensione così ottenuta viene raddrizzata da D3 e stabilizzata sui 15V dal diodo Zener D2.
La rete formata da R4 e C3 ha lo scopo di limitare le interferenze indotte dalla commutazione del triac.
Qualora si impiegassero triac ad elevate correnti di innesco, è possibile diminuire il valore di R3 portandolo a 150 Ohm.

IL FOTOACCOPPIATORE

Il più semplice fotoaccoppiatore è costituito da una coppia LED/Fototransistor in cui il LED, polarizzato da un segnale esterno, illumina il fototransistor che entra in conduzione

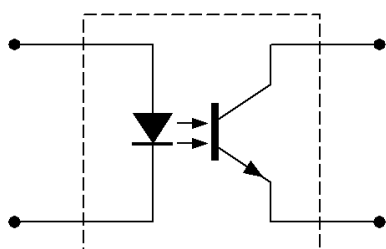



Figura 2 Simbolo elettrico di un fotoaccoppiatore

Diodo led	
Corrente max continua	50÷60mA
Corrente max di picco	2÷3A
Potenza max dissipata	100mW
Tensione di lavoro media	1÷1,5V
Corrente di lavoro media	10mA
Tensione di rottura inversa	5÷7V
Capacità di giunzione	30÷80pF
Transistor ricevente	
Tensione max collettore/emettitore	30÷70V
Corrente max collettore	50mA
Potenza dissipata	100÷200mW
Velocità di commutazione	4÷10μs
Isolamento	
Tensione di isolamento	1000÷5000V
Capacità di accoppiamento	0,3÷1 pF

Figura 3 Caratteristiche elettriche tipiche dei fotoaccoppiatori più comuni

(figura 2). Per evitare interferenze con sorgenti di luce esterna, il tutto è racchiuso in un contenitore plastico opaco. Di solito i fotoaccoppiatori garantiscono un isolamento da 1000 a 5000V ed alcuni modelli arrivano fino a 15000V.

Per comodità nella tabella di figura 3 sono riportate le caratteristiche dei fotoaccoppiatori più comuni composti da una coppia led/fototransistor, mentre nella figura 4 sono riportate le sigle e le relative caratteristiche di comuni fotoaccoppiatori.

Come pilotare un fotoaccoppiatore

Come visto in figura 3, il diodo di ingresso può sostenere correnti fino ad un massimo di 60mA, dunque per pilotare correttamente un fotoaccoppiatore non si deve superare tale limite.

SIGLA	Isolamento	Guadagno transistor	Vce_{MAX}
4N25	2500	50	30
4N26	1500	20	30
4N27	1500	10	30
4N32	6000	400	50
4N33	6000	400	50
4N35	3500	100	30
4N36	2500	100	30
4N37	1500	100	30
CNV17	4400	180	70
CNV17F	5300	120	70
CNV75	5300	100	80
CQV80	4400	80	35
IL250	5000	100	30
IL.CA230	6000	100	30
IL.CA255	6000	100	55
ILD.1	6000	50	30
ILD.74	6000	35	20
IL.CT6	6000	35	30
ILQ.1	6000	20	30
ILQ.74	6000	15	20
MCP.3020	7500	-	400
SFH600	2800	100	70
SFH601	5300	100	70
SFH609	5300	100	90
SFH610	2800	180	70
SFH611	2800	180	70

Figura 4 Alcuni tipi di fotoaccoppiatori e relative caratteristiche

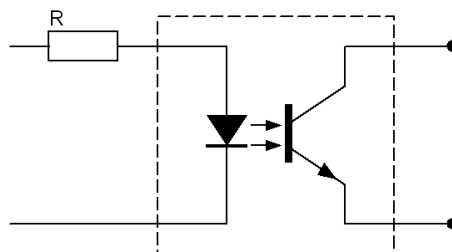



Figura 5 Per limitare la corrente, si deve inserire una resistenza in serie al diodo

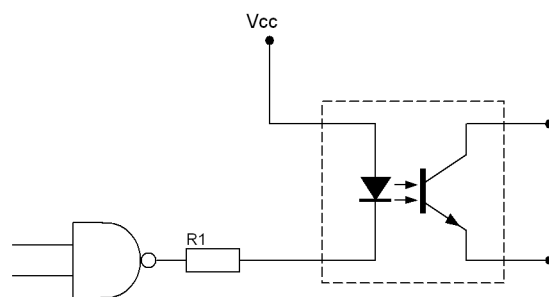



Figura 6 Pilotaggio di un fotoaccoppiatore sul livello basso della porta logica

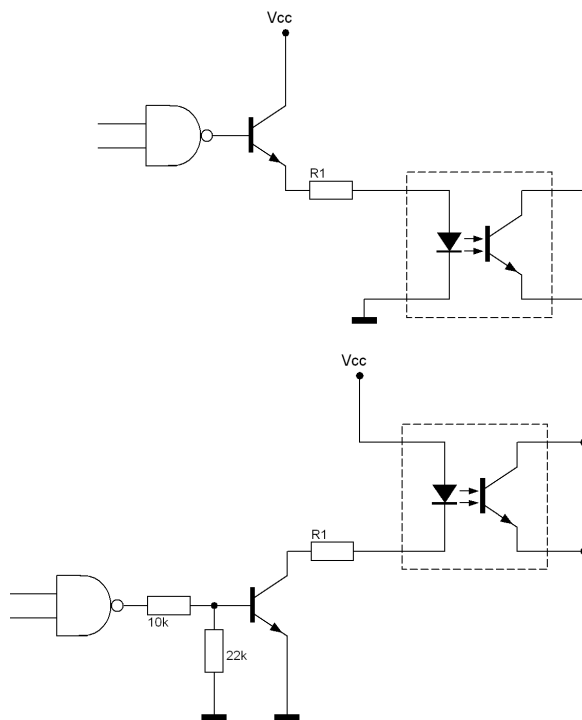



Figura 7 Due schemi equivalenti per pilotare il fotoaccoppiatore sul livello alto

È buona norma mantenere la corrente di ingresso sui valori di 10÷15mA. Per limitare la corrente nel diodo, si usa una resistenza limitatrice in serie al diodo stesso come mostrato in figura 5. Il valore di tale resistenza può essere calcolato come:

$$R=(Vcc-Vd)/Io$$

dove Vcc è la tensione massima di pilotaggio, Vd la caduta di tensione sul diodo (si può assumere Vd=1,2V circa) e I_o il valore di corrente fissato per il diodo.

Nel caso in cui si debba pilotare il fotoaccoppiatore con l'uscita di una porta TTL si deve tener conto che queste porte non erogano grandi correnti sul livello alto, dunque sarà necessario collegarle come mostrato nella figura 6. In questo caso il diodo sarà polarizzato sul livello logico basso in uscita dalla porta.
Anche in questo caso il valore di R1 viene calcolato con la formula già vista. Se si intende polarizzare il diodo sul livello alto, si deve utilizzare un transistor NPN che fornirà la corrente necessaria. Il transistor deve essere connesso secondo uno degli schemi di figura 7.

Qualora sia necessario trasferire segnali analogici può essere necessario pilotare il fotoaccoppiatore con un amplificatore operazionale. Lo schema è quello di figura 8 e, come si può notare, è stato aggiunto un diodo D1 in parallelo al diodo emettitore.
Lo scopo di tale diodo è quello di evitare che tensioni negative raggiungano il fotoaccoppiatore provocando il danneggiamento del fotoaccoppiatore stesso.

Per il calcolo della resistenza si può utilizzare la formula R1=(Vm-1,2)/0,015 dove Vm è il valore massimo di tensione in uscita all'amplificatore.

Come prelevare il segnale da un fotoaccoppiatore

Normalmente il segnale di uscita di un fotoaccoppiatore non è un segnale ben squadrato e pulito, per cui, soprattutto se si devono pilotare circuiti logici, è consigliabile, inserire un inverter Trigger di Schmitt in modo da ripulire il segnale. Il circuito è quello di figura 9 in cui la resistenza di pull-up da 10KOhm garantisce il livello logico alto qualora il transistor sia interdetto.

È possibile, in alternativa usare il transistore interno come pull-up ed inserire una resistenza esterna di pull-down come mostra la figura 10. Se l'inverter è un TTL si dovrà usare un resistenza da 180 Ohm, se l'inverter è un TTL-LS si dovrà scegliere una resistenza da 560 Ohm, mentre se l'inverter è un CMOS si può scegliere un valore compreso tra i 1000 ed i 10K Ohm.

Pilotaggio di relé e motori

Per eccitare un relé lo schema da utilizzare è quello di figura 11. In questo caso l'uscita del fotoaccoppiatore pilota un transistore NPN

Figura 8 Pilotaggio mediante amplificatore operazionale

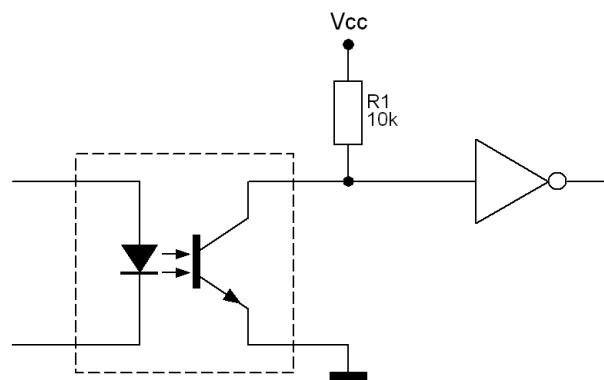



Figura 9 Pilotaggio di un inverter con pull-up

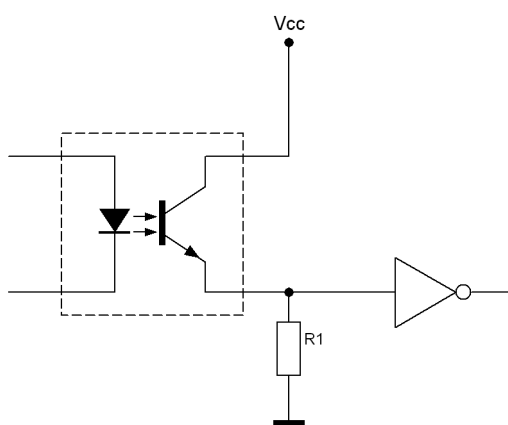



Figura 10 *Pilotaggio di un inverter con pull-down*

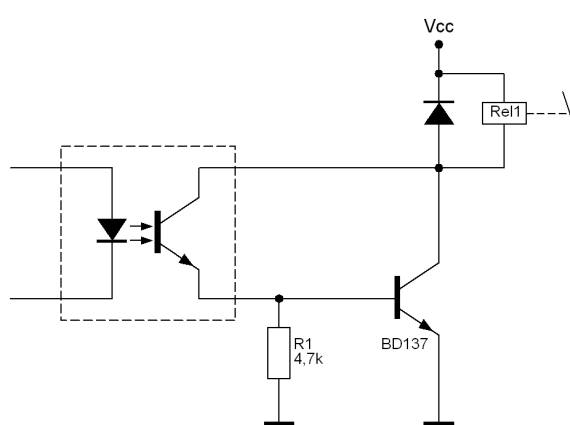



Figura 11 *Pilotaggio di un relé*

(ad esempio un BD137) il quale, se in saturazione, provoca l'eccitazione di un relè.
Il diodo in parallelo al relé garantisce la protezione del transistor durante la sua commutazione.
Qualora si intenda pilotare un carico in alternata (si esso un motore o una lampada) si può usare lo schema di figura 12.

Figura 12 *Pilotaggio di un carico in alternata*

Utilizzando un transistor BD137, la massima corrente sopportabile è 0,5A e può essere aumentata utilizzando un transistore più potente.
Come tensione di alimentazione è bene non superare i 28-29Vac.

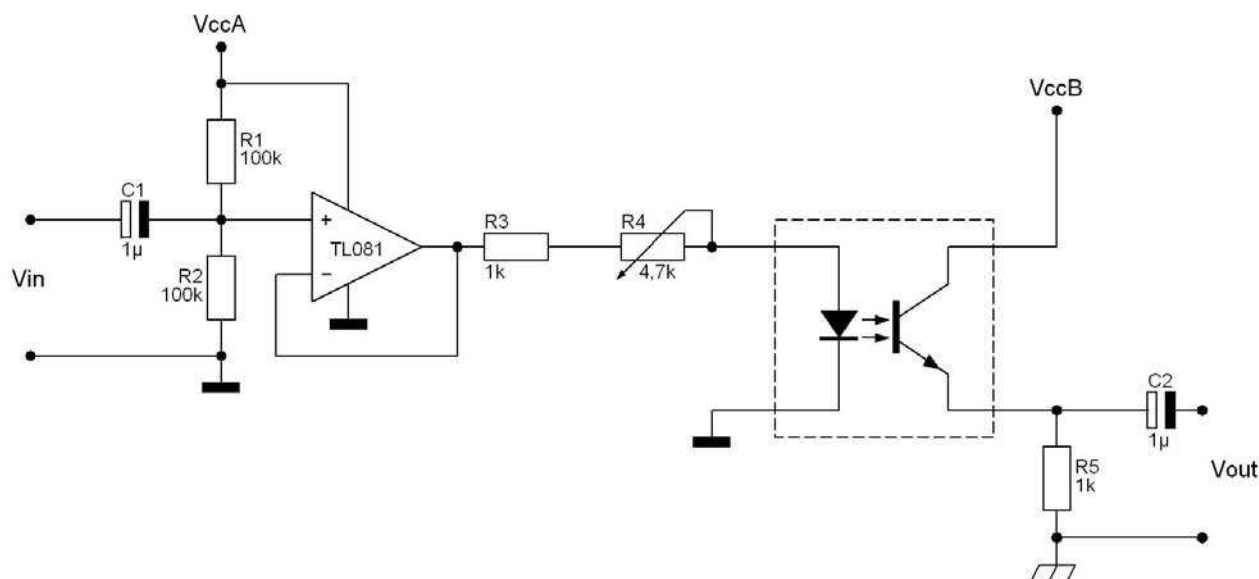



Figura 13 *Trasferimento di segnali lineari in BF*

Trasferimento di segnali lineari

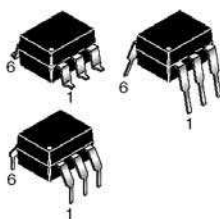
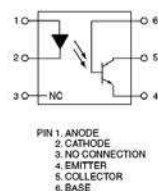
Un fotoaccoppiatore non si presta molto al trasferimento di segnali lineari in quanto il diodo di ingresso provoca grandi distorsioni. Tuttavia lo schema proposto in figura 13 consente questa particolare applicazione e, con i valori indicati, è possibile trasferire segnali nella banda 20Hz-20KHz senza alcuna distorsione.

R1 ed R2 consentono di sommare al segnale di ingresso una tensione continua in modo che l'uscita sia sempre positiva.

L'unica taratura da fare sul circuito è regolare il trimmer R4 in modo da ottenere ai capi di R5 una tensione pari alla metà della tensione di alimentazione VccB. Il condensatore C2 elimina poi tale componente continua dal segnale di uscita.

GENERAL PURPOSE 6-PIN PHOTOTRANSISTOR OPTOCOUPLERS

4N25	4N26	4N27	4N28	4N35	4N36
4N37	H11A1	H11A2	H11A3	H11A4	H11A5

BLACK PACKAGE (NO -M SUFFIX)

SCHEMATIC

ABSOLUTE MAXIMUM RATINGS (T_A = 25°C unless otherwise specified)

Parameter	Symbol	Value	Units
TOTAL DEVICE			
Storage Temperature	T_{STG}	-55 to +150	°C
Operating Temperature	T_{OPR}	-55 to +100	°C
Wave solder temperature (see page 14 for reflow solder profiles)	T_{SOL}	260 for 10 sec	°C
Total Device Power Dissipation @ T_A = 25°C	P_D	250	mW
Derate above 25°C		3.3 (non-M), 2.94 (-M)	
EMITTER			
DC/Average Forward Input Current	I_F	100 (non-M), 60 (-M)	mA
Reverse Input Voltage	V_R	6	V
Forward Current - Peak (300µs, 2% Duty Cycle)	I_F(pk)	3	A
LED Power Dissipation @ T_A = 25°C	P_D	150 (non-M), 120 (-M)	mW
Derate above 25°C		2.0 (non-M), 1.41 (-M)	mW/°C
DETECTOR			
Collector-Emitter Voltage	V_{CEO}	30	V
Collector-Base Voltage	V_{CBO}	70	V
Emitter-Collector Voltage	V_{ECO}	7	V
Detector Power Dissipation @ T_A = 25°C	P_D	150	mW
Derate above 25°C		2.0 (non-M), 1.76 (-M)	mW/°C

I CD-ROM di FareELETTRONICA

FareELETTRONICA

Annata 2003 e 2004

22 Uscite di Fare Elettonica in 2 CD-ROM!

Due annate complete in formato ACROBAT (PDF).

Potrai sfogliare comodamente tutte le riviste uscite nel 2003 e nel 2004 e stampare gli articoli di tuo interesse.

I CD delle annate sono acquistabili separatamente oppure insieme ad un prezzo scontato.

Vitamina C

Il corso completo Vitamina C in formato ACROBAT (PDF)

Tutte le lezioni del corso base al **linguaggio C** comodamente consultabili e stampabili in alta risoluzione.

Notevolmente migliorata e completata rispetto alla versione originale pubblicata sulla rivista.

Il **CD-ROM Vitamina C** contiene una sezione "**Contenuti Speciali**" tutta da scoprire.

PIC® microcontroller by example

PIC® Microcontroller By Example

Il corso completo PIC® Microcontroller By Example in formato ACROBAT (PDF)

Tutte le lezioni pronte per la consultazione con i sorgenti dei progetti immediatamente utilizzabili nelle tue applicazioni.

Il modo migliore per avere sempre sottomano la soluzione per il tuo progetto con i **PICmicro®**.

Il **CD-ROM PIC® Microcontroller By Example** contiene una sezione "**Contenuti Speciali**" tutta da scoprire.

Ordinali subito su www.farelettronica.com oppure telefona allo 02.66504794

Inware Edizioni - Via Cadorna, 27/31 - 20032 Cormano (MI)

Vitamina C: Algoritmi di

In questa puntata verranno presentati alcuni importanti algoritmi di uso comune, per i quali confronteremo diverse possibilità di implementazione e le loro prestazioni. Gli esempi di codice riportati serviranno anche ad approfondire alcuni aspetti avanzati del linguaggio C.

INTRODUZIONE

In ogni programma esistono alcune funzioni che vengono richiamate più frequentemente di altre nel corso delle elaborazioni, che cioè costituiscono il "nucleo" del programma o di un certo algoritmo. In alcuni casi queste funzioni possono essere richiamate perfino più volte all'interno di uno stesso loop. E' chiaro che in questi casi il modo in cui le funzioni sono state scritte può incidere notevolmente sui tempi di esecuzioni del programma stesso. Questo aspetto, che molto spesso è sottovalutato nell'implementazione di software "applicativo", diventa particolarmente importanti in quei casi in cui il programma controlla un sistema che deve necessariamente reagire in tempi brevi, o che deve elaborare una grande quantità di dati.
Queste considerazioni possono sembrare banali, ma andrebbero sempre fatte quando ci si accinge a scrivere un programma, in quanto i loro effetti a lungo termine o in presenza di una grande quantità di dati possono essere tutt'altro che trascurabili! (Alcuni esempi numerici riportati di seguito rendono meglio l'idea). Per ottenere funzioni efficienti (dal punto di vista del tempo di esecuzione) bisogna considerare due aspetti differenti, ma entrambi molto importanti: l'ottimizzazione del codice, e l'utilizzo di algoritmi di complessità computazionale minima. In questa puntata ci focalizzeremo soprattutto sul secondo aspetto, applicato ad una classe di funzioni di uso molto generale, e per la quale esistono degli importanti risultati anche da un punto di vista teorico. Dalla prossima puntata invece considereremo il primo aspetto, con particolare riferimento alla scrittura di routine matematiche/aritmetiche.

ALGORITMI DI RICERCA

Un algoritmo di ricerca è, come suggerisce il nome, un algoritmo che ha lo scopo di ricercare una particolare sequenza all'interno di un set di dati più o meno grande. Questa esigenza è molto generale, e si può presentare in programmi con compiti molto diversi, da applicazioni come i database, a programmi che svolgono elaborazioni di dati più a basso livello, in cui si possono utilizzare delle tabelle di consultazione o di indicizzazione. Normalmente queste tabelle possono essere utilizzate ad esempio per registrare la presenza di determinati dati, o per contenere informazioni aggiuntive associate a ciascuno di essi.
In molti casi è possibile trattare l'insieme di dati su cui si esegue la ricerca come un array contenente un numero arbitrario di elementi. In questi casi l'implementazione degli algoritmi di ricerca risulta abbastanza semplice e soprattutto veloce, in quanto gli elementi (non necessariamente i dati) sono disposti secondo un preciso ordine, ed è possibile accedevi direttamente in base al loro indice.
L'algoritmo di ricerca più semplice che si può adottare in questi casi è quello di **"ricerca linea-**

di Antonio Di Stefano
a.distefano@farelettronica.com

ricerca ed ordinamento

re", che è anche il più intuitivo, e consiste nello scorrere gli elementi dell'array a partire dal primo, fino ad incontrare (eventualmente) l'elemento ricercato.
Il codice seguente implementa l'algoritmo di ricerca lineare in un ipotetico programma che riceve in ingresso un codice binario a 32 bit e fornisce in uscita un altro codice consultando un'apposita tabella in cui sono riportate le coppie ingresso-uscita. Questo funzionamento è simile a quello utilizzato in alcuni file system per reperire la posizione dei file sul disco, in agenti intelligenti dotati di un comportamento stimolo-risposta, ed in tantissimi altri casi.

```
#include <stdio.h>

// *** definizioni ***
#define N_ELEM 128

typedef struct {
  int input;
  int output;
} Dato;

// *** prototipi ***
void inizializza_tabella(Dato *tabella,int n);
int ricerca(int inp, Dato *tabella,int n);

// *** main ***
void main()
{
  Dato tabella[N_ELEM];
  int i, o;

  inizializza_tabella(tabella, N_ELEM);

  printf("Ingresso: ");
  scanf("%d", &i);
  o=ricerca(i, tabella, N_ELEM);

  printf("Output: %d\n", o);
}

// *** Funzione: ricerca lineare ***
int ricerca(int inp, Dato *tabella, int n)
{
  int i;

  for(i=0; ((tabella[i].input!=inp)&&(i<n));
i++){}

  if (tabella[i].input!=inp)
  {
    // dato non trovato
    return 0;
  } else {
    // dato trovato
    return tabella[i].output;
  }
}
```

Nell'esempio sia gli ingressi che le uscite sono rappresentati da numeri interi a 32 bit, quindi ogni elemento dell'array conterrà una coppia ingresso-uscita. A tal proposito è stato definito il tipo strutturato "Dato" (tramite l'istruzione **typedef**). La tabella è dichiarata all'interno della funzione main, ed è costituita semplicemente da un array di elementi di tipo Dato. Il numero massimo di elementi della tabella in questo caso è fisso, comunque in ogni momento la tabella può anche essere compilata soltanto in parte (ulteriori elementi possono essere aggiunti man mano che questi si rendono disponibili).
La tabella viene inizializzata da una funzione apposita, di cui non è mostrato il codice, e che chiaramente dipende dalla particolare applicazio-

ne. La funzione di ricerca vera e propria invece viene invocata una volta ottenuto il dato d'ingresso, utilizzando come parametri, oltre all'elemento di input per il quale si vuole ottenere l'uscita corrispondente, il puntatore alla tabella (ossia all'array), ed il numero di elementi contenuti nella tabella stessa. La funzione compie le operazioni descritte prima, che sono implementate semplicemente con un ciclo **for** "modificato" (si sarebbe potuto usare anche un più tradizionale **while**). Il ciclo non ripete nessuna operazione, ma ha il solo scopo di incrementare l'indice i fino a quando l'ingresso fornito non risulta uguale ad uno degli ingressi memorizzati nella tabella o non si supera il numero di elementi memorizzati. A questo punto il ciclo si interrompe conservando il valore dell'indice. Il successivo controllo verifica se è stato effettivamente trovato l'elemento, oppure se il ciclo è arrivato fino alla fine della tabella senza trovare alcuna corrispondenza. In quest'ultimo caso la routine restituisce un valore di default. Se si vuole evitare questo, ed invece evidenziare meglio questa condizione occorre restituire un valore apposito, ad esempio un numero che sicuramente non è contenuto nei dati ricercai. Oppure restituire l'indice dell'elemento piuttosto che il suo valore, e ad esempio usare valori negativi per segnalare eventuali errori.
Riguardo al programma possono essere fatte alcune considerazioni, ad esempio: si potrebbe pensare che dal momento che sia i dati in ingresso che quelli in uscita sono dei valori numerici, sarebbe possibile usare direttamente l'indice della tabella per reperire il valore cercato, senza eseguire una ricerca. Quando in effetti è possibile solo quando il numero elementi è relativamente piccolo (per esempio nel caso di ingressi ad 8 bit, ai quali corrispondono soltanto 256 combinazioni), oppure quando è possibile usare una funzione che riesca a convertire valori "sparsi" in una sequenza più o meno ristretta di valori consecutivi (funzione di "**hash**").
In generale invece è necessario utilizzare un algoritmo di ricerca vero e proprio. Un'altra osservazione che è possibile fare riguarda la funzione di confronto adottata, che in questo caso è rappresentata dall'uguaglianza numerica ("="). In generale l'operatore di confronto da utilizzare dipende sia dal tipo di dati, sia dai particolari criteri che si vogliono adottare, che dipendono dal problema specifico.
Riguardo all'algoritmo utilizzato invece è possibile notare che la sua "**complessità computazionale**" è di ordine n (da qui il nome di "ricerca lineare"). Questo significa che il tempo richiesto da una ricerca è direttamente proporzionale al numero di elementi presenti nella tabella (considerando il caso peggiore). Questa è una proprietà della modalità di ricerca adottata, non tanto dell'implementazione: è il problema stesso della ricerca (formulato in questo modo) che ha una sua intrinseca complessità minima, che non è possibile aggirare con nessun algoritmo. Se abbiamo 1 milione di dati, potrebbe essere necessario scorrerne proprio 1 milione prima di trovare quello giusto!
Tuttavia, se il problema lo permette, è possibile introdurre degli elementi che rendano la ricerca più facile e veloce. L'idea di base è che se i dati sono ordinati secondo un certo criterio, è possibile sfruttare proprio il loro ordinamento per trovare più velocemente quello che ci interessa (se esiste). Una delle più note tecniche di ricerca che si basa su questo principio è la "**ricerca binaria**". Il procedimento è abbastanza semplice: supponiamo di disporre di n dati, disposti secondo un ordine crescente; consideriamo il dato che si trova nella posizione centrale e poi, a seconda che questo sia maggiore o minore di quello cercato consideriamo la metà di dati superiore o inferiore e ripetiamo il procedimento continuando a dividere in maniera iterativa gli intervalli in due (come in una specie di metodo di bisezione). Alla fine per trovare (o non trovare) il dato cercato saranno necessari al massimo $\log_2 n$ operazioni anziché n! La formula non rende bene l'idea, ma significa che nello stesso caso di prima, cioè per 1 milione di dati, occorrerebbero soltanto 20 operazioni!
È chiaro quindi che quando è possibile utilizzare una ricerca binaria è conveniente farlo. Per fare questo è necessario preordinare i dati, oppure inserirli direttamente in ordine.
Il codice riportato di seguito mostra come modificare la routine di ricerca vista prima per implementare una ricerca binaria:

```
// *** funzione: ricerca binaria ***
int ricerca(int inp, Dato *tabella, int n)
{
  int i=0;

  do {
    n=n/2;
    if (tabella[i].input>inp)
       i=i-n;
    else
       i=i+n;
   } while ((n>0)&&(tabella[i].input!=inp));

  if (tabella[i].input!=inp)
  {
    // dato non trovato
    return 0;
  } else {
    // dato trovato
    return tabella[i].output;
  }
}
```

La routine somiglia alla precedente, la differenza principale risiede nel fatto che invece di spostarsi in posizioni consecutive dell'array, in questo caso ci si sposta su e giù ad intervalli di $n/2^m$ (metà, un quarto, un ottavo…). Questo è ottenuto considerando un indice i che inizialmente parte dalla posizione 0, a cui è sommato o sottratto un termine che è ottenuto dividendo iterativamente per 2 la lunghezza dell'array. La somma o la sottrazione sono decisi in base al risultato del confronto tra l'elemento corrente e quello cercato. Per esempio, se supponiamo che gli ingressi memorizzati nella tabella siano dei numeri consecutivi compresi tra 0 e 127, e quello cercato è l'85, i valori assunti da n e da i saranno i seguenti:

Risposta al quesito codice LST23702

L'uscita vale -14V con ingresso nullo a causa della tensione di offset che si comporta come un generatore in continua applicato all'ingresso. In questo caso è come avere un generatore di 1mV all'ingresso dell'integratore. Applicare una continua ad un circuito integratore permette di avere in uscita una rampa che porta l'operazionale alla saturazione. L'uscita dell'integratore in funzione del tempo e con una tensione Vo continua all'ingresso è del tipo:

$$Vu=-Vot/RC$$

(nell'ipotesi che il condensatore sia inizialmente scarico). Con 1mV in ingresso (dovuto alla tensione di offset) e con i valori scelti di R e C, l'equazione della rampa di uscita è Vu=-100t e questa raggiunge il valore di -14 per t=140ms.

Il vincitore di Febbraio 2005 (LST23601)
I nostri complimenti a **Marco Perolio** di Borgosesia (VC)
che vince un abbonamento a Fare Elettronica!

```
i    n
64,  64
96,  32
80,  16
88,  8
84,  4
86,  2
85,  1
```

se non si trova l'elemento cercato il ciclo termina quando n raggiunge il valore 0. Una cosa importante da notare è che questo metodo funziona bene quando il numero di dati è una potenza di 2 (in questo caso 128). Se questo non è verificato esistono degli accorgimenti per continuare ad utilizzarlo ugualmente, oppure è possibile utilizzare altri metodi simili. Tra questi i più pratici sono quelli che consistono ad esempio nel raggruppare in blocchi i dati in base a delle caratteristiche comuni, e conservare l'indice corrispondente a ciascuno di questi blocchi (ad esempio dati relativi a giorni dell'anno potrebbero essere raggruppati per mese, o delle stringhe per la lettera iniziale). In questo caso la ricerca richiede al massimo un numero di iterazioni pari alla lunghezza del blocco, quindi con questo sistema si hanno delle buone prestazioni, anche se è necessario l'uso di alcuni puntatori aggiuntivi (che possono essere conservati in un secondo array). Questi algoritmi si rivelano particolarmente utili nel caso di **dati organizzati in liste**, in cui non è possibile applicare tecniche simili alla ricerca binaria, a causa dell'impossibilità di leggere direttamente un elemento a partire da un indice (la lista deve sempre essere percorsa dall'inizio).

ALGORITMI DI ORDINAMENTO

Una seconda importante classe di algoritmi è rappresentata dagli algoritmi di ordinamento. La loro importanza deriva dal fatto che oltre ad essere impiegati in molte applicazioni comuni, sono spesso utilizzati in combinazione con gli algoritmi di ricerca per aumentarne l'efficienza, come visto prima.
La applicazioni degli algoritmi di ordinamento sono molto più ampie di quello che si potrebbe pensare inizialmente, infatti il concetto di "ordinamento" deve essere inteso in senso molto generale, esso non è limitato al caso di ordine numerico o alfabetico, ma può estendersi a casi del tutto arbitrari. Infatti, così come nella ricerca i criteri di "uguaglianza" erano stabiliti in base al tipo di dati ed all'operatore adottato per eseguire il confronto, anche nell'ordinamento è possibile utilizzare dei criteri assolutamente arbitrari. Si pensi ad esempio agli algoritmi di scheduling utilizzati dai sistemi operativi per eseguire in time-sharing i diversi processi e task, o per gestire l'accesso alle risorse condivise: la scelta di una particolare sequenza implica necessariamente un ordinamento, che deve tenere conto di fattori come priorità, stato attuale, tempi di attesa, timeout, etc. Escludendo l'operatore di ordinamento gli algoritmi risultano comunque indipendenti dal particolare problema trattato.
Ci riferiremo di seguito al caso di ordinamento di un array, sia per maggiore semplicità, sia perché molti casi pratici sono riconducibili a questo.
Esistono molti metodi per ordinare gli elementi di un array, uno dei più semplici ed intuitivi è il seguente (supponiamo di dovere ordinare per valori "crescenti" l'array): consideriamo il primo elemento e scorriamo il resto dell'array fino a trovare un elemento con valore minore o uguale, quindi scambiamo i due valori; consideriamo quindi il successivo e ripetiamo per ciascuno la stessa procedura. Quando giungeremo all'ultimo valore, l'array sarà ordinato. Il codice che implementa questo algoritmo, per i dati considerati nel programma di esempio precedente, è riportato di seguito:

```
void ordina(Dato *tabella, int n)
{
  int i, j, min;
  Dato temp;

  for(i=0; i<n; i++)
  {
    min=i;
    for(j=i; j<n; j++)
      if
(tabella[j].input<=tabella[min].input)
min=j;
```

```
    temp=tabella[i];
    tabella[i]=tabella[min];
    tabella[min]=temp;
  }
}
```

Il codice ordina i dati della tabella in base al valore crescente del campo "input". Data la loro natura numerica l'operatore di confronto utilizzato è il semplice "minore o uguale" ("<="). Il primo **for** è utilizzato per determinare l'indice di partenza, il secondo per cercare il valore minimo nella parte restante dell'array. Va notato che l'indice del secondo **for** inizia dal valore attuale del primo. Questo significa la parte "bassa" dell'array ad ogni passo risulta sempre ordinata.
Un altro famoso algoritmo di ordinamento è il cosiddetto "**bubble sort**", che esegue quasi le stesse operazioni, ma risulta leggermente più semplice ed elegante:

```
#define SWAP(a,b)   {int t; t=a; a=b; b=t;}

void bubble(Dato *tabella, int n)
{
  int i, j;

  for(i=0;i<n;i++)
  {
    for(j=1;j<(n-i);j++)
    {
      if(tabella[j-
1].input>tabella[j].input)
        SWAP(tabella[j-
1].input,tabella[j].input);
    }
  }
}
```

L'algoritmo scorre l'array n volte scambiando tra loro i due elementi adiacenti, se questi non risultano ordinati nel modo voluto.
Si può notare che in entrambi gli algoritmi vengono utilizzati due cicli **for** annidati. Questo ci fa capire che per ordinare n elementi, è necessario eseguire approssimativamente nxn operazioni! La complessità degli algoritmi di ordinamento semplici è infatti dell'ordine di n^2. Per dati di grandi dimensioni questo può essere un fattore molto limitante: se ad esempio abbiamo un array di 1000 elementi, potrebbero essere necessarie 1milione di operazioni!
Per fortuna, come vedremo nel prossimo paragrafo, esiste una soluzione buona e molto comoda a questo problema. Prima di concludere però è il caso di soffermarci un attimo sull'applicazione degli algoritmi di ordinamento visti ad insiemi di dati costituiti da **liste**. È facile riconoscere che un algoritmo come il bubble sort si presta bene ad essere applicato anche a liste, infatti sebbene non sia possibile accedere direttamente ai singoli elementi, è possibile conservare i puntatori che interessano (in particolare quello relativo all'elemento adiacente), e quindi ricondursi ad una situazione simile a quella degli array.

L'ALGORITMO QUICKSORT

Per superare il limite della complessità quadratica degli algoritmi semplici di ordinamento, è possibile utilizzare un algoritmo noto come Quicksort, inventato da C. A. Hoare. Questo algoritmo è abbastanza complesso (e per questo motivo non verrà spiegato qui), ma ha il vantaggio fondamentale di riuscire a ordinare un insieme di dati con un numero di operazioni dell'ordine di $n \log_2 n$ nella maggior parte dei casi (per dare un'idea nel caso di 1000 elementi occorrono circa 10000 operazioni). Il Quicksort utilizza una tecnica di tipo "*divide et impera*", cioè scompone ricorsivamente l'intero insieme di dati in parti più piccole, che vengono ordinate separatamente in modo più semplice. Un'ulteriore vantaggio di questo algoritmo, oltre al costo computazionale, è che esso è disponibile come funzione nelle librerie standard dell'ANSI C! Per la precisione la funzione in questione di chiama **qsort** e risiede nella libreria **stdlib.h**. Il prototipo della funzione riserva alcune sorprese, ed a prima vista potrebbe lasciare un pò perplessi:

```
void qsort(void *base, size_t nelem, size_t
width, int (*fcmp)(const void *, const void
*));
```

Nonostante la lunghezza della dichiarazione i parametri da passare alla funzione sono solo 4. Il primo (*base*) è il puntatore al primo elemento dell'array da ordinare (ossia il nome dell'array stesso), il secondo (*nelem*) è un intero che definisce quanti sono gli elementi da ordinare, il tezo (*width*) indica la dimensione in byte di ciascun elemento dell'array, ed il quarto, cioè int (**fcmp*)(*const void *, const void **) è **il prototipo di una funzione!** Non è molto noto (ne tanto mento è una possibilità sfruttata spesso), ma in C è possibile passare ad una funzione un'altra funzione come parametro! Questa possibilità ha delle potenzialità enormi, a cui non accennerò in questo articolo, ma che lascio comunque alla fantasia del lettore. Nel caso della funzione qsort la funzione da passare è quella che ha lo scopo di eseguire il confronto in base al quale si eseguirà poi l'ordinamento dei dati. Come abbiamo visto prima i dati da ordinare non hanno necessariamente una natura numerica o alfabetica, e quindi possono richiedere delle funzioni apposite per stabilire il significato di "precedere" o "seguire" (o similmente "minore" o "maggiore") all'interno di un insieme ordinato.

La funzione qsort richiede una funzione che prenda i puntatori ai due elementi da confrontare e restituisca un valore intero che vale:

< 0 se *elem1 "<" *elem2
= 0 se *elem1 "=" *elem2
> 0 se *elem1 ">" *elem2

Dove il simbolo "<" significa "precede", ">" significa "segue", e "=" invece "uguale/equivalente".

Quindi, per utilizzare la funzione qsort è necessario scrivere un'altra funzione che esegua il confronto tra due elementi, e passarla come parametro. Facciamo subito un esempio: consideriamo il programma visto in precedenza ed il tipo di dati già utilizzato, e supponiamo di volere ordinare in modo crescente i dati contenuti nell'array (in base al valore del campo .input). Dal momento che il campo .input è di tipo numerico, per ottenere i valori voluti basta eseguire una sottrazione (ad esempio se elem1<elem2 allora elem1-elem2 sarà < 0, e così via). La funzione di confronto sarà la seguente (il nome della funzione è arbitrario):

```
int qcmp(Dato *a, Dato *b)
{
    return (a->input)-(b->input);
}
```

A questo punto sarà sufficiente richiamare la funzione qsort con i seguenti parametri:

```
qsort(tabella, N_ELEM, sizeof(Dato), qcmp);
```

Da notare che la dimensione degli elementi dell'array è stata ricavata utilizzando l'istruzione **sizeof**, mentre la funzione è stata passata semplicemente indicandone il nome. Questo è in accordo con quanto dichiarato nel prototipo della funzione qsort, perché in effetti i nomi delle funzioni in C, così come avviene nel caso degli array, sono dei puntatori (!), e precisamente indicano l'indirizzo da cui inizia il codice ad esse associato.

Nel caso abbastanza frequente che sia necessario eseguire un confronto tra stringhe al fine di ordinarle alfabeticamente è possibile utilizzare la funzione **strcmp** della libreria **string.h:**

```
int strcmp(const char *s1, const char *s2);
```

che restituisce un parametro direttamente utilizzabile dalla funzione qsort. Se si vuole ottenere un ordine alfabetico inverso basta anteporre il segno – al risultato restituito della funzione.

CONCLUSIONE

Dal prossimo articolo verrà affrontato un argomento molto importante ed interessante dal punto di vista applicativo: l'implementazione efficiente di algoritmi matematici in C. In particolare verrà mostrato come implementare algoritmi basati su aritmetica floating point utilizzando solamente l'aritmetica intera (molto più veloce ed adatta alla programmazione di piccoli sistemi di calcolo), e come utilizzare efficientemente gli operatori aritmetici e bitwise.

Le so tutte!!!

Rispondi correttamente al quiz e parteciperai all'estrazione di un abbonamento omaggio a

FareELETTRONICA

Partecipare è semplicissimo:

rispondi al quesito seguendo il regolamento e, se la risposta si rivelerà esatta, parteciperai all'estrazione di un abbonamento omaggio (o al rinnovo qualora fossi già abbonato) a Fare Elettronica. Ogni mese sulle pagine della rivista troverai la soluzione del quesito del mese precedente e il nome del vincitore di uno dei quesiti pubblicati. Per tutti i partecipanti è previsto comunque un coupon del 10% di sconto utilizzabile per un acquisto sullo shop di www.farelettronica.com.

QUESITO codice LST23803

Chiave: Una rete Bluetooth

Individuate la chiave leggendo da sinistra a destra le caselle contrassegnate.

Orizzontali:

1 Logica ad emettitori accoppiati.
2 Condensatore incognito.
3 Mezzo triac.
4 Entrata, ingresso.
5 Diodo ad emissione di luce.
6 Applicato ad una resistenza da 1 Ohm, si ottiene una corrente di 1A.
7 Sigla di vari apparecchi professionali Sommerkamp.
8 Simbolo dell'unità di misura del campo elettromagnetico "H" nel sistema cgs.
9 Può essere un dipolo, uno stilo, ma anche un sistema molto complesso di elementi che "risuonino".
10 Genere di Gate.
11 Può essere da 1/10W, ma anche da diversi KW.
12 Dispositivo che serµve a bilanciare un sistema di captazione sbilanciato e viceversa: in genere ha ingressi ed uscite a 75 Ohm e 300 Ohm.
13 Oscillatore controllato in tensione.
14 Quella dei ricevitori si misura in µ/m (abbreviato).
16 Transistore ad effetto di campo.
17 Alta tensione rialzata.
18 Negli impianti HI-FI commerciali, questo valore non di rado è… gonfiato.
19 Una volta, al posto di questo diodo, si usavano tubi OA2, OB2, VR150, STV250/30 e simili.
20 Funzione trigonometrica.
21 Un dispositivo che "parla" (abbreviato).

Verticali:

1 Si definisce così, familiarmente, un noto costruttore USA di IC, calcolatori e orologi digitali.
2 Può avere un gran numero di poli
3 Una porta logica.
4 I tre inventori di questo dispositivo, presero nel 1965 il Premio Nobel per la Fisica.
5 Può essere a carbone, a filo e perfino a liquido.
8 Metallo che ha la caratteristica di venire a mancare, presso gli sperimentatori, quando i negozi sono chiusi o è giorno di festa.
9 Modulazione che è contenuta, con quella video, nel segnale TV.
10 Si conosce quello di Wheatstone, di Waterloo, quello di Wien…
11 Operazione logica.
21 Misurano notevoli correnti elettriche.
22 Un bus seriale universale.

Scadenza: il termine ultimo per rispondere è il 30 Aprile 2005

Regolamento

1) Il quiz è aperto a tutti i lettori.
2) Saranno considerate esclusivamente le risposte pervenute entro la scadenza indicata nel quesito.
3) Inviare la risposta compilando il modulo su www.farelettronica.com/lesotutte oppure via **fax** allo 02/66508225 o tramite **lettera** a Inware Edizioni, via Cadorna 27- 20032 Cormano (MI), citando il codice del quesito.
4) **Indicare nella risposta il tuo nome, cognome, indirizzo, telefono ed eventuale indirizzo email.** Non ci sarà possibile considerare le risposte con **dati mancanti o insufficenti.**

Riflettori su...

NEMO10:

Si chiama NEMO10 ed è l'ultimo arrivato in casa Sena Technologies. Consente la connessione ad una rete LAN attraverso una porta seriale e la sua particolare compattezza (45x45x18,5 mm) lo rende adatto all'integrazione su dispositivi embedded come modulo on-board.

Nemo10 è un modulo a basso costo pensato per utenti OEM che intendono includere la connettività Ethernet direttamente nella propria applicazione. L'interfaccia a 24 pin dual-in-line di NEMO10, riduce il tempo ed i costi di realizzazione del layout del circuito stampato sul quale verrà inserito, consentendo immediatamente la connessione con una rete LAN.

Dal lato seriale il modulo supporta comunicazioni fino a 115.2Kbps in TTL con i segnali Rx, Tx, RTS, CTS, DSR e DTR. Sul lato Ethernet dispone di una interfaccia 10Base-T e supporta indirizzi IP statici o dinamici. Sono disponibili anche segnali per il pilotaggio di led di stato e switch per il reset ed il ripristino delle impostazioni di fabbrica.

NEMO10 può essere alimentato a 5V con un consumo di 0,3W e può essere configurato attraverso la porta seriale o da remoto mediante un collegamento telnet.

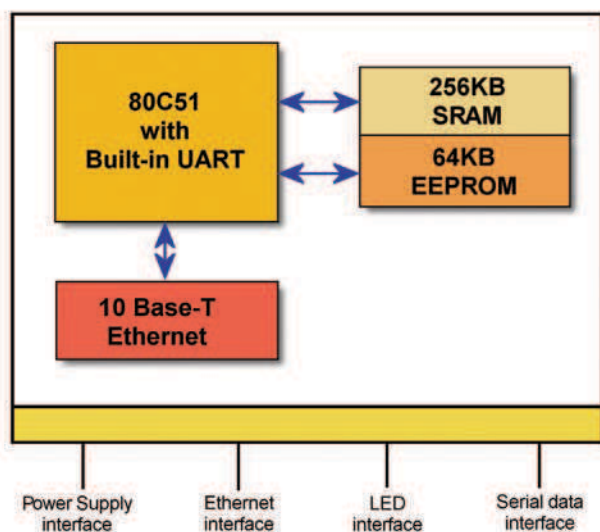



***Figura 1** Schema a blocchi del modulo NEMO10*

Pin No.	JP1
1	Vcc
2	Console/Data switch
3	Serial DSR
4	Serial CTS
5	Serial DTR
6	Serial RTS
7	Serial Rx
8	Serial Tx
9	Factory Reset switch
10	GND
11	LED (Serial Tx)
12	12

Pin No.	JP2
1	GND
2	Reset
3	LED (Ethernet Tx)
4	LED (Ethernet Rx)
5	LED (Ethernet Link/Collision)
6	Ethernet Tx-
7	Ethernet Tx+
8	Ethernet Rx+
9	Ethernet Rx-
10	Vcc
11	LED (Ready)
12	LED (Ethernet Act)

***Figura 2** Segnali ai pin del modulo NEMO10*

Un convertitore seriale/Ethernet piccolo e robusto

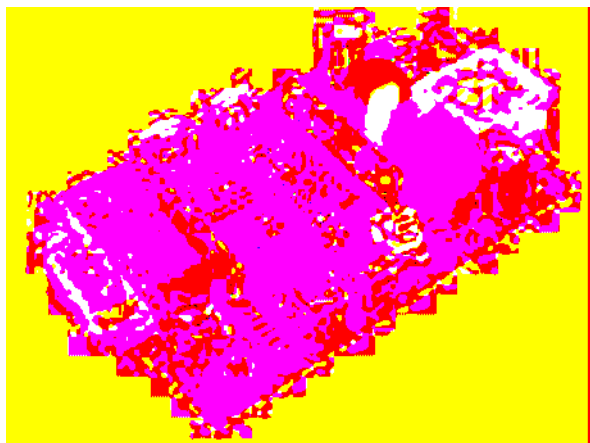



Figura 3 NEMO10 Starter KIT

GLI STRUMENTI

Per la configurazione del dispositivo è possibile usare la porta seriale come porta console o connettersi da remoto via telnet. In alternativa Sena Technologies fornisce una applicazione Windows che permette di impostare facilmente tutti i parametri di configurazione.

Per lo sviluppo dell'hardware è disponibile uno starter kit (figura 3) comprendente un modulo NEMO10 ed una board sulla quale può essere inserito. La board dispone di un connettore DB9 per il collegamento seriale, un connettore RJ45 per il collegamento LAN, il regolatore di tensione per l'alimentazione e tutti i led e switch necessari al corretto funzionamento.

Informazioni

Per informazioni su NEMO10:
Elettroshop
Via Cadorna, 27/31 – 20032 Cormano (MI)
Tel. 02.66504794 Fax 02.66508225
info@elettroshop.com – www.elettroshop.com

Hardware	Security
80C51 8-bit microprocessor, 32KB SRAM, 64KB EEPROM	Password-protection
Ext. Interface pins	Management
24-pin dual in line for UART, Ethernet, LED and Power	Telnet, HelloDeivice Manager, console port
Serial Interface	Power
Built-in UART. Up to 115.2Kbps. 7/8 data bits Signals: Rx, Tx, RTS, CTS, DTR, DSR, GND	5Vdc ± 10%, 60mA@5Vdc
Network Interface	Environmental
10 Base-T Ethernet	Operating temp.: 5°C to 50°C
Protocols	Form Factor
ARP, IP/ICMP, TCP, Telnet, DHCP, PPPoE	45x45x18,5 mm

Riflettori su...
Gli appuntamenti di Fortronic 2005

Una tre giorni ricca di momenti di incontro per l'electronic community che si riunisce a Padova dal 21 al 23 aprile 2005.

Si parlerà di innovazione, produzione e distribuzione a Fortronic, la fiera dell'elettronica industriale a Padova dal 21 al 23 aprile. Le tre tematiche saranno sviluppate attraverso convegni, workshop e seminari tecnici.

GLI INCONTRI DELL'INNOVAZIONE

Competitività e innovazione è il tema di apertura del Forum. Il finanziamento alle imprese innovative (h. 10.00 sala B, pad.8) è l'evento organizzato da Veneto Innovazione, per mettere in evidenza I meccanismi e gli strumenti finanziari a supporto delle aziende start up. Per capire metodologie e strumenti operativi a disposizione delle aziende, Unindustria e Parco Scientifico e Tecnologico Galileo organizzano un convegno dal titolo Dall'Idea al Prodotto (h. 15.30 sala A, pad. 8).
Sempre nella giornata di giovedì si parlerà di progettazione innovative focalizzando l'attenzione sulle tecnologie emergenti nei Sensori Wireless (h. 14.30 sala B, pad 8) e si metteranno a confronto ASIC strutturati e FPGA (h. 11.00 sala A, pad 8). Una tavola rotonda (h. 15.00 sala C, pad. 8) vedrà invece svilupparsi un dibattito tra le aziende del settore dei sistemi di visione.

IL TEMA DEL SENZA PIOMBO

Cem, Ems, Odm e tecnici saranno impegnati venerdì 22 aprile in una serie di convegni.
In particolare un convegno dedicato al senza piombo (h. 10.00 sala B, pad.8) cercherà di traccare la roadmap verso il 2006, in vista delle nuove direttive europee sulle sostanze pericolose presenti nelle apparecchiature elettroniche.
E ancora Direttive RAEE al traguardo, (h.14.30 sala A, pad 8) metterà a confronto la politica ambientale italiana con esperienze dall'estero.

MERCATO E DEMAND CREATION

È dedicato alla distribuzione il convegno di IDEA che si terrà sabato 23 aprile: il XII International Distribution Meeting (h. 10.00 sala B, pad 8) dal titolo "Economics in a changing global economy - Future trends in distribution" tratterà i differenti aspetti della Demand Creation. Sarà dedicato invece a Buyer e Credit Manager l'appuntamento in programma In cui si parlerà di condizioni di acquisto.

Fortronic – Electronics Forum si svolge dal 21 al 23 aprile 2005, presso i padiglioni 5 e 8 di FieraPadova.

Per avere informazioni specifiche circa la manifestazione è possibile telefonare al numero verde **800-320216**.

Per ricevere la tessera d'ingresso gratuita è sufficiente registrarsi on-line all'indirizzo **www.fortronic.it**.